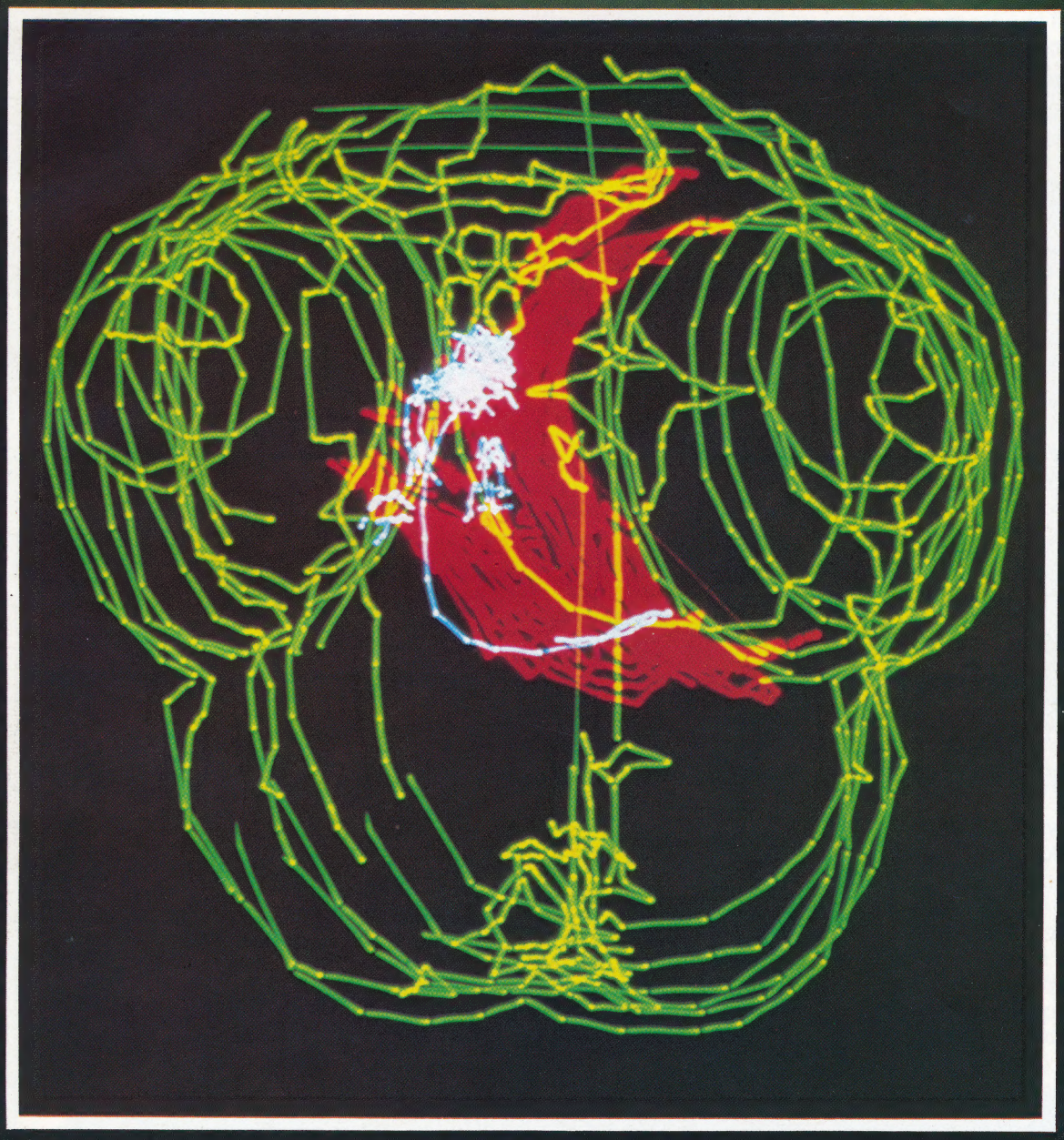
I.T.E.C.



Information
Technology·
Electronics·
Computers·

47 in weekly parts

THE HOME STUDY COURSE IN ELECTRONICS AND COMPUTERS

U.K. 99p Australia $2.50 New Zealand $3.00 South Africa R2.75

Contents Part 47

I.T.E.C. is complete in 51 weekly parts (50 parts + Index).

POSTAL SUBSCRIPTIONS

UK readers
You can arrange to receive weekly copies of *I.T.E.C.* at 99p each inclusive of postage and packing. Simply calculate the cost of your subscriptions by multiplying the number of parts you require by 99p and send a cheque or postal order for that amount to:–

Subscriptions Dept., GEJ Publishing Ltd
187 Oxford Street, LONDON W1R 1AJ.

with a note of your name and address, where you wish the copies to be sent and the part number from which you want the subscription to start. Please allow 28 days for your subscription to be processed plus postal time.

Eire readers
As above, but you should calculate the cost of your subscription at IR£1.75 per part.

Other overseas readers
May pay in the currency of their choice provided that its value when converted to sterling is £1.50 per part. Minimum subscription 10 parts.

CONSULTANTS

This series has been produced in collaboration with **TEXAS INSTRUMENTS** Learning Centre. Editorial and graphics assistance is gratefully acknowledged.

U.K. Technical Consultant
Dr. Robert King, Reader in Communications Engineering, Imperial College of Science and Technology, London.

BINDERS

U.K. & Eire
Details of how to obtain your binders for *I.T.E.C.* will appear in part 50.

Australia
Write to *I.T.E.C.* Binders, Gordon & Gotch Ltd., PO Box 213, Alexandria, New South Wales 2015.

New Zealand
Write to *I.T.E.C.* Binders, Gordon & Gotch (NZ) Ltd., PO Box 1595, Wellington

South Africa
Binders are available at R7.95 (please add sales tax) from any branch of Central News Agency. In case of difficulty please write to Intermag, P.O. Box 57934, Springfield 2137.

Other countries
Binders are available from your local newsagent.

BACK NUMBERS

U.K.
Back numbers can be ordered from your newsagent or from *I.T.E.C.* BACK NOS., Whinfrey Strachan Ltd., 187 Oxford Street, London W1R 1AJ. Price 99p each inc. p + p. Please allow 21 days for delivery.

Eire
As for U.K. but please remit in Sterling if ordered by post.

Other countries
Copies available from your local newsagent.

INDEX

The final part of *I.T.E.C.* will be a fully cross-referenced INDEX to the complete work

Editor:
Amanda Kirk

Technical Editor:
Keith Brindley

Art Editor:
Alistair Carlisle

Sub Editor:
Nicholas Bellenberg

Copy Consultant:
Chris Wallace (Creative Technical Services)

Published by GEJ Publishing Ltd.

Distributed by Whinfrey Strachan Ltd.
187 Oxford Street, London W1R 1AJ

Printed in England by Southernprint Ltd.

Photographs, unless otherwise stated, are from the library of Texas Instruments.

Cover: a computer generated reconstruction of a fish embryo. The 'head' is shown in green and muscle is shown in red.
(Photo: Science Photo Library/Dr R B Fishman).

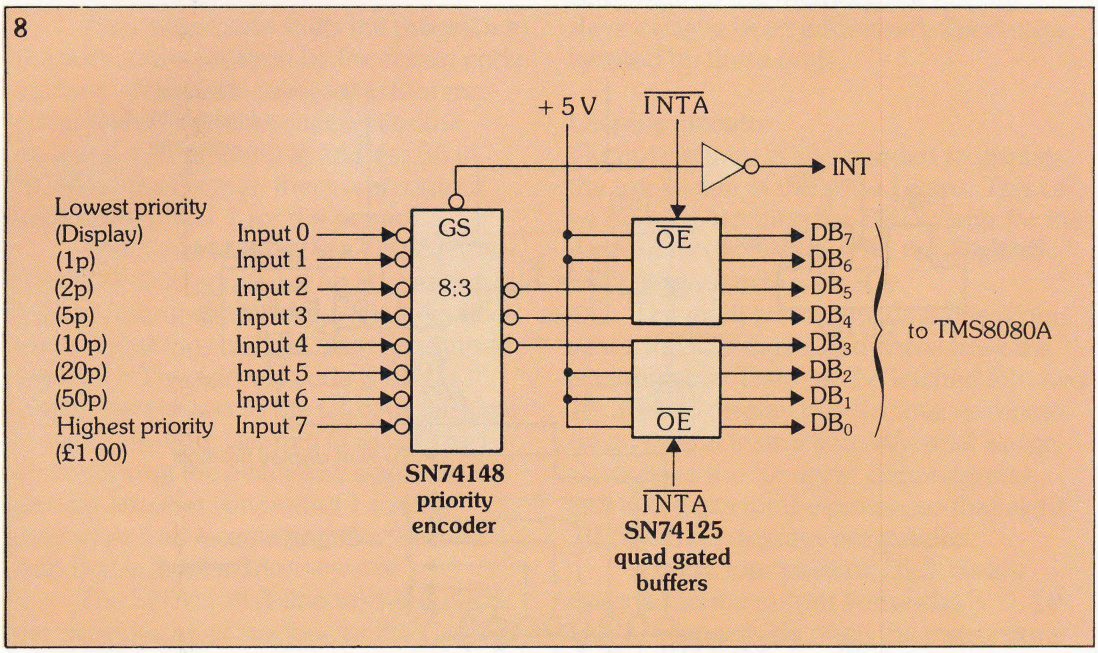
(continued from part 46)

An application for the '8080A

We'll assume that a mechanical device has been built, that sorts 1p, 2p, 5p, 10p, 20p, 50p and £1.00 coins by denomination. The sorted coins are then separately channelled, by denomination, to automatic packagers, and as they move down each coin is identified and its value noted and added to a total.

The system uses optoelectronic sensors to identify the coins so it must, therefore, respond to interruptions of the light beam, by adding an appropriate amount to the total. It should also provide a switch to command a display of the total sterling value of the sorted coins. The switch is also used to turn off the sorting machine.

8



8. Restart generating circuitry for the TMS8080A.

Hardware design

Remember that the '8080A can handle eight interrupt signals, and that there are eight different restart instructions. Since we are dealing with seven different types of coin, seven different light beams are required to identify them. In turn, this means that seven different restart instructions can be generated (as shown in *figure 8*).

The switch is also used to generate an eighth restart instruction. So, a display switch causes a subroutine jump (RST 8) to location 64 in memory; a penny detection jumps to location 56, and so on. The corresponding inputs, interrupts and jump locations are shown in *table 1*. RST 0 is not used, you'll have noticed, as only a RESET signal causes a jump to memory location 0.

The subroutine programs that service the interrupts either cause the appropriate amount to be added to the accumulating total (when a coin causes the interrupt) or, if the display switch is activated, cause the total to be displayed and the sorter stopped. The $\overline{\text{INTA}}$ gates the generated restart code onto the data bus en route to the microprocessor.

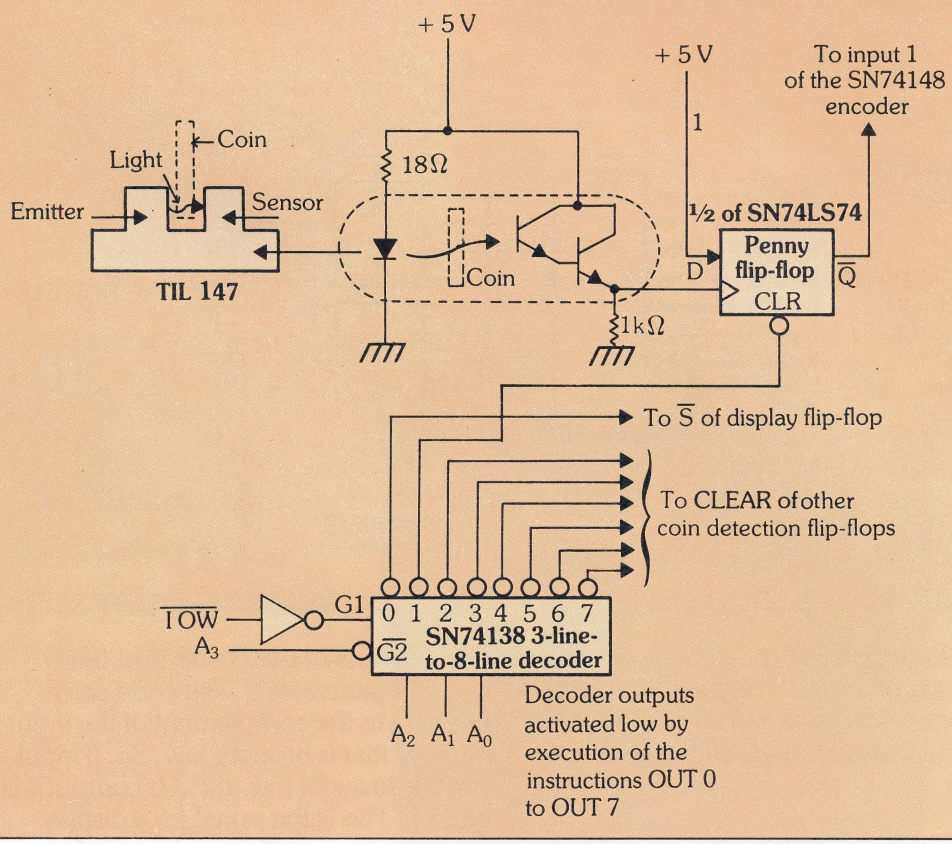
The SN74148 8-line-to-3-line encoder generates a 3-bit code corresponding to the complement of the input line number that is brought low. So, if input line 0 is brought low, the 3 bit output will be 111. This is the signal for a display request interrupt. If line 6 is brought low, on the other hand, the output will be 001, for RST 2 (50p detection) and so on. The SN74148 is a priority encoder, with input line 7 (£1.00 detection) having the highest priority (000).

The INT signal to the TMS8080A is received from the GS output of the encoder, and this output is low every time an input low signal is received.

Table 1
Inputs, interrupts and subroutine jumps

Input line	Input function	Interrupt (RST n)	Location (n × 18)	Priority encoded output
0	Display	RST 8	64	111
1	1p	RST 7	56	110
2	2p	RST 6	48	101
3	5p	RST 5	40	100
4	10p	RST 4	32	011
5	20p	RST 3	24	010
6	50p	RST 2	16	001
7	£1.00	RST 1	8	000

9



9. Detection circuits for each denomination of coin are identical – this is the penny detection circuit.

Input interface circuits

The restart generating circuitry shown in *figure 8* has to develop the input signals and hold them low, until the microprocessor responds. A TIL147 optoelectronic source and Darlington sensor assembly can be used to detect the presence of a coin (*figure 9*). Remember, each denomination of coin is sent down a different channel after sorting, so each channel has its own sensor.

Figure 9 shows the penny detector, which generates an input 1 for the SN71418 priority encoder each time a penny passes through the beam of light. The other coin detectors are duplicates of this circuit.

As a coin passes the slot of the

TIL147 and cuts off the light beam, the transistor current changes from 4 mA to 0 mA – this sends a 0 (low) to the CLOCK input terminal of the D flip-flop, 74LS74. When the coin passes out of the slot, the transistor current is restored and a 1 goes to the D flip-flop's clock terminal. This sets the output Q to 1 and $\overline{Q}$ to 0. The active low on $\overline{Q}$ is applied to input 1 of the SN74148, which provides the INT signal and the restart code to the '8080A, at the correct time.

In the case of the display flip-flop, an active low from the switch is applied to the $\overline{R}$ terminal of an $\overline{S}$–$\overline{R}$ flip-flop to reset Q to 0. This low from Q is fed to input 0 of the SN74148 and generates the restart code (*figures 8* and *11*).

The restart code shifts the program to the subroutine location by the restart code address. Within the subroutine that performs the operations to recognise the detected coin (starting at address 56 in ROM for the penny), there is an output instruction (OUT 1 for the penny).

When executed by the TMS8080A, this instruction sends out an address code and an output enable signal that identifies the fact that the coin has been recognised (*figure 9*). The address code is used to send an active low to the CLEAR terminal of the D flip-flop, clearing Q to 0, $\overline{Q}$ to 1, and releasing the active low signal on the priority encoder (on output 1 for the penny) so that it can recognise the next coin that is detected.

The SN74138 3-line-to-8-line decoder provides an active low for the CLEAR terminal, as shown in *figure 9*. It detects the address code and the enable signals produced by the OUTPUT operation and outputs the CLEAR signal on the appropriate line. G1, an active high enable for the decoder, is provided by $\overline{IOW}$ and is ANDed with an active low enable to G2 provided by the address bit A_3. In this way, the decoder is made to respond to the output instruction addresses 0 to 7 when an output instruction is executed.

As you can see, output line 1 is connected to the D flip-flop. The decoder's other outputs are connected to the other flip-flops' appropriate CLEAR terminals. For example, OUT 2 clears the 2p flip-flop; OUT 4 the 5p flip-flop; and OUT 7 clears the £1.00 flip-flop.

The display flip-flop is reset to an uninterrupted condition by OUT 0, but in a different way. In this case, the active low on line 0 goes to the $\overline{S}$ terminal of the display flip-flop to reset the Q output to a 1, releasing the input 0 of the priority encoder. After the coin detector and display switch detector flip-flops are cleared or set back, they wait for a new coin or switch closure detection.

In each case, the circuits detect the execution of an appropriate output instruction by the TMS8080A and use this information to determine which flip-flop to clear. During this execution, the accumulator data is placed on the data line, but since neither the memory nor display devices have been addressed, the data is ignored by these units.

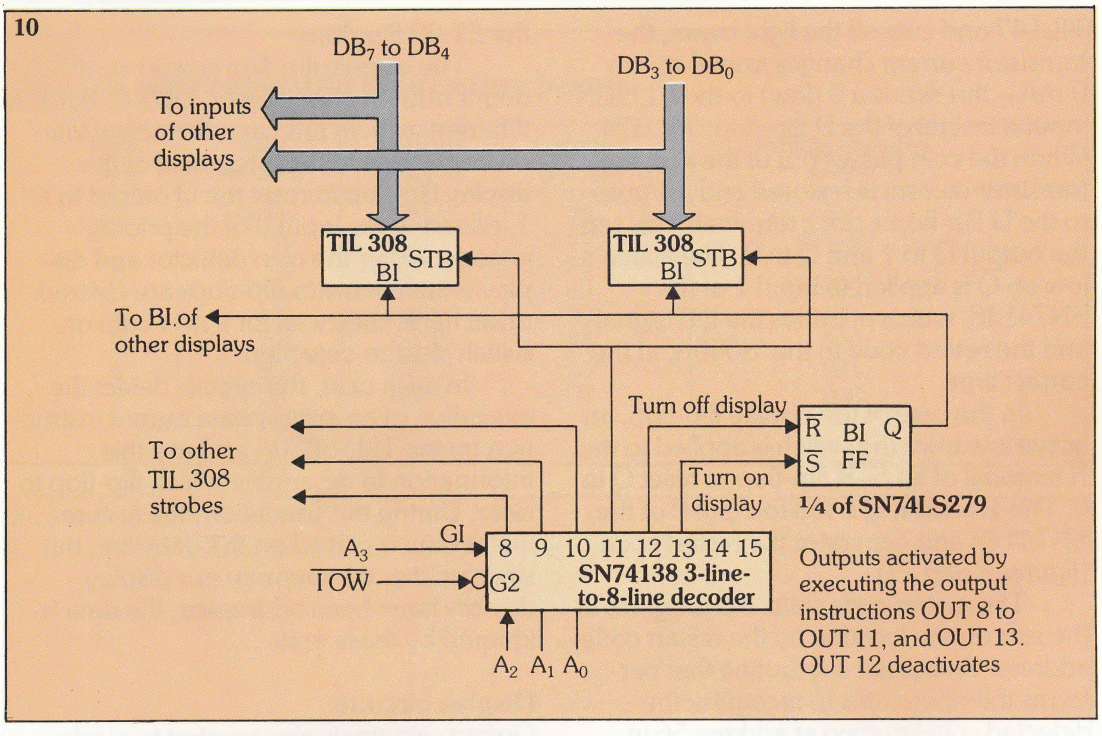
Display circuits

Output circuitry is also needed to display the total value of the sorted coins. This can be done by latching the BCD codes for the digits to be displayed, onto the TIL308 LED displays with logic ICs.

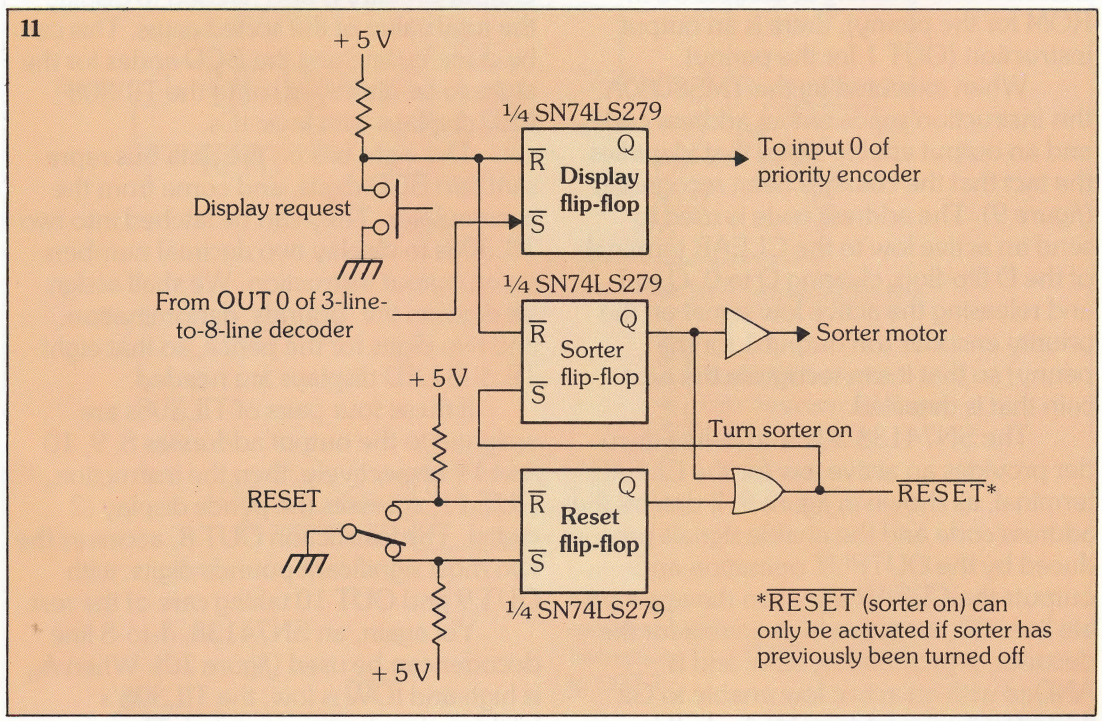
The eight bits on the data bus represent two BCD digits, and come from the accumulator. They can be latched into two TIL308s to display two decimal numbers on an output instruction. We shall assign six digits to the 'pounds' denomination, and two digits for the pence, so that eight TIL308 LED displays are needed.

If these four pairs of TIL308s are assigned to the output addresses 8, 9, 10 and 11 respectively, then the instruction, OUT 11, accesses the pence display (2 digits). The instruction OUT 8, accesses the two most significant pounds digits, with OUT 9 and OUT 10 taking care of the rest.

Yet again, an SN74138, 3-to-8 line decoder can be used (*figure 10*). When A_3 is high and $\overline{IOW}$ is low, the TIL308's latching signals are generated. Thus, it responds to the address A_0 to A_3 during the execution of an output instruction. When an output instruction with the address 1011 is executed (as a result of an OUT 11 instruction), A_3 will be high, $\overline{IOW}$ will be low, and output pin 11 of the SN74138 will also be low, latching the available data to be displayed onto the two TIL308s used to show the pence data.



10. Output circuitry to display the total value of sorted coins.



11. Display interrupt and sorter control connections.

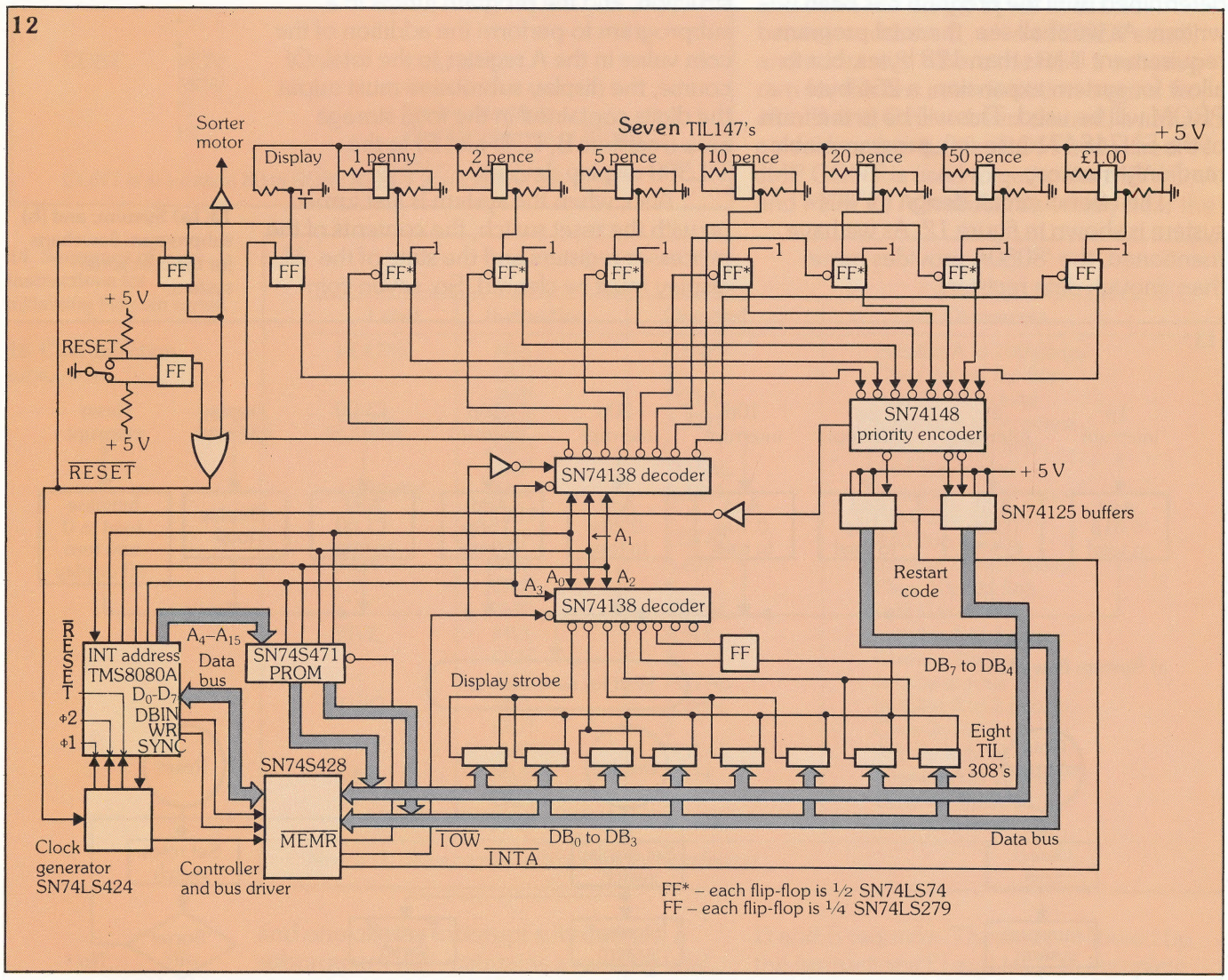
Corresponding address codes latch data for the other three pairs of TIL308s.

The entire display can be turned on and off by controlling the signal applied to the TIL308's blanking inputs (BI). So, as shown in *figure 10*, an OUT 12 instruction (12 being the address code assigned to turn off the display) makes the decoder deliver a 0 to the $\overline{R}$ terminal of the BI flip-flop. This, in turn, causes Q and the BI line to be low and turn off the display. When an OUT 13 instruction is executed, the display is turned on. In this case, the decoder output sends a 0 to the $\overline{S}$ terminal of the BI flip-flop to set Q and the BI line to a 1, turning on the display.

The request for a display is signalled by pushing a momentary contact switch

which resets the Q output of the display flip-flop to an active low (*figure 11*). Since this is connected to input 0 of the SN74148 priority encoder, as shown in *figures 8* and *12*, a RST 7 is generated causing a subroutine branch to location 56 in ROM. If an OUT 0 instruction in that subroutine is executed, the output of the first SN74138 decoder sets the display flip-flop Q output back to 1.

12. Overall circuit design for the coin sorter system.



Reset sequence and sorter control

The display request switch can also be used to turn the sorter off, if it is used to control the sorter flip-flop as shown in *figure 11*. When display is requested, the sorter flip-flop output Q goes low (0) and turns the sorter motor off.

The Q output of the sorter's flip-flop is fed into an OR gate with the reset signal. If the sorter is on, Q is a 1 and reset cannot be generated. With Q at 0 (sorter off), an active low can be generated on the reset line. When the reset switch is pushed, a momentary signal is generated to force the TMS8080A to begin executing instructions at location 0 in ROM. This reset condition also sets the sorter flip-flop output to a 1 to turn the sorter on again.

Memory design

Only four bytes of memory are needed to store the 8-digit coin value total and the six 8-bit registers inside the '8080A can be used for this purpose. If the system had to use a more complicated program, then additional RAM would be needed to save information held in the stack registers

whenever a subroutine was required.

However, as the coin sorter uses a program structure that is totally dependent on interrupts, the stack information does not have to be saved, so no external RAM is needed.

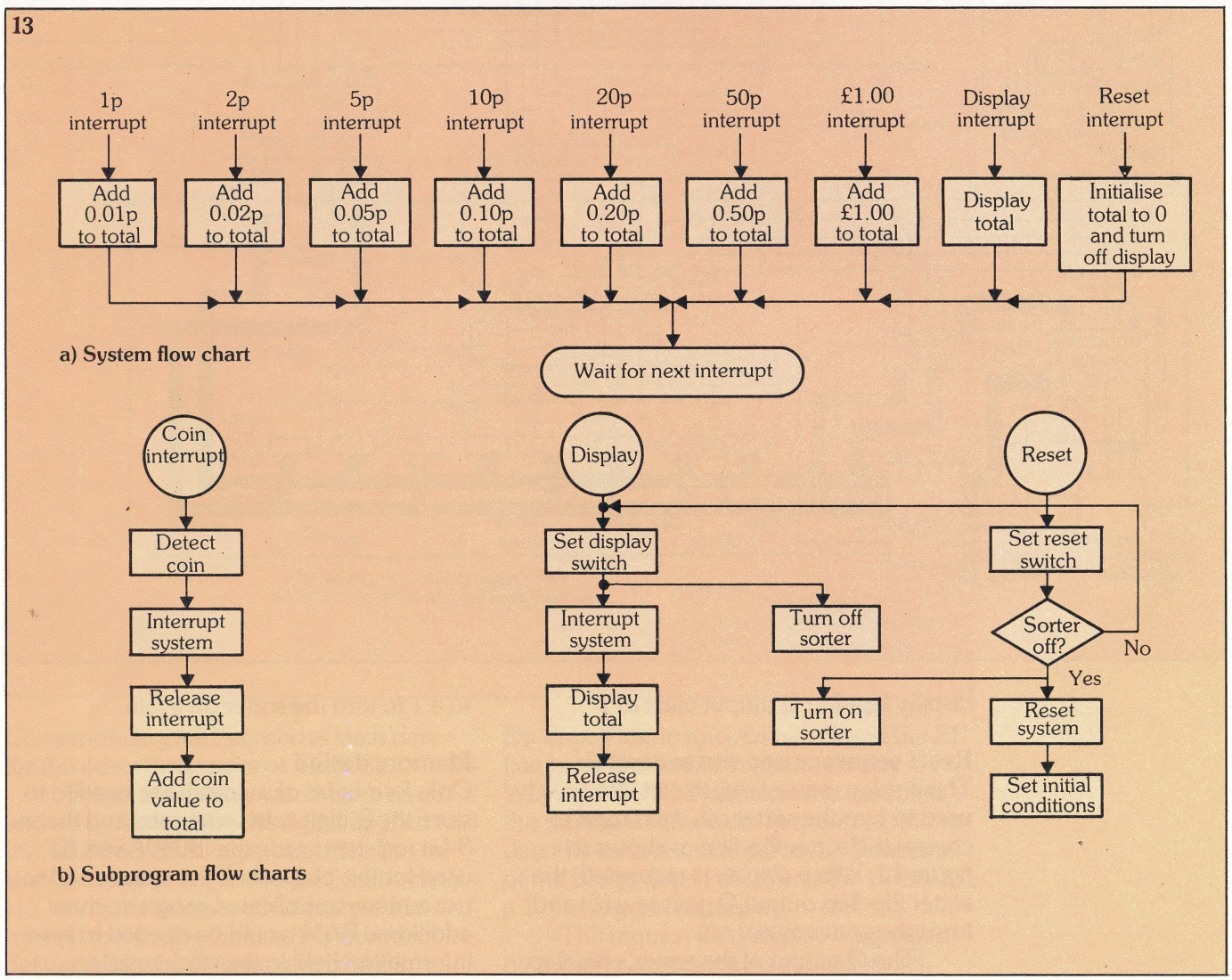
The ROM memory design depends on the number of bytes required to hold the program. This, of course, cannot be determined until the program has been written. As we shall see, the total program requirement is less than 128 bytes, but to allow for system expansion, a 256 byte PROM will be used. This will be in the form of the SN74S471 fuse-link programmable read only memory.

The overall circuit design for this system is shown in *figure 12*. As we have mentioned, the '8080A provides more than enough data registers.

Developing the program

The coin sorter's flowchart is shown in *figure 13*. Each restart or interrupt subroutine simply loads the A register with the coin value to be added to the accumulating sterling total stored in registers B, C, D and E. A signal is then output to the coin flip-flop clear terminal with an OUT instruction, and the program jumps to a subprogram to perform the addition of the coin value in the A register to the total. Of course, the display subroutine must ouput the digits contained in the total storage area (registers B, C, D and E) to the TIL308 display devices.

Also, when the system is first turned on with the reset switch, the contents of the processor registers and the state of the display must be cleared. So, seven coin



13. (a) System; and (b) subprogram flowcharts for the coin sorter system.

14

Label	Instruction	Operand	Comments
RESET	OUT	12	Turn off display
	XRA	A	Clear accumulator
	MOV	B,A	Clear B
	MOV	C,A	Clear C
	JMP	CONT	Go to rest of sequence
	•		
	•		
	•		
	•		
CONT	MOV	D,A	Clear D
	MOV	E,A	Clear E
	EI		Enable interrupts
	HLT		Wait for interrupt or RESET

CONT is at location 80 in ROM; RESET is at location 0 in ROM.

14. Sequence of instructions for initialising the coin sorter.

15. Coin detection subroutines.

15

Label	Instruction	Operand	Comments
POUND	MVI	A,64H	Initialise A to 100 pence
	OUT	7	Clear pound flip-flop
	JMP	TOTAL	Add £1.00 to total
FIFTY	MVI	A,32H	Initialise A to 50 pence
	OUT	6	Clear 50p flip-flop
	JMP	TOTAL	Add 50p to total
TWENTY	MVI	A,14H	Initialise A to 20 pence
	OUT	5	Clear 20p flip-flop
	JMP	TOTAL	Add 20p to total
TEN	MVI	A,AH	Initialise A to 10 pence
	OUT	4	Clear 10p flip-flop
	JMP	TOTAL	Add 10p to total
FIVE	MVI	A,5H	Initialise A to 5 pence
	OUT	3	Clear 5p flip-flop
	JMP	TOTAL	Add 5p to total
TWO	MVI	A,2H	Initialise A to 2 pence
	OUT	2	Clear 2p flip-flop
	JMP	TOTAL	Add 2p to total
ONE	MVI	A,1H	Initialise A to 1 penny
	OUT	1	Clear 1p flip-flop
	JMP	TOTAL	Add 1p to total

Note: an H after a number indicates that it is hexadecimal

and one display interrupt subroutines, a subprogram that maintains a running total, and a sequence to set the initial conditions must be written.

Setting initial conditions

When the reset switch causes the program counter to clear to a ROM address of zero (the 'reset' flowchart in *figure 13*), the first instruction shown in *figure 14*, OUT 12, turns off the total display. Then the A register is cleared with the XRA A instruction (when a register is XORed with itself the result is all zeros in the register). B to E registers are cleared by moving the A register contents (now all zeros) into them.

At the end of this sequence, the interrupt signal is enabled with the EI instruction and an HLT (halt) instruction causes the processor to wait for an interrupt (or another RESET). Unfortunately, this sequence requires 9 bytes of ROM, and there are only 8 bytes available between address 0, where RESET starts, and address 8, where the RST 1 subprogram for the first coin detection begins. So, after B and C have been cleared, a jump to a new location (80) in ROM is required. A JMP CONT is used to get to the MOV D, A and MOV E, A instructions which clear the D and E registers. These are followed by the instructions EI and HLT to complete the sequence.

Coin detection

All the coin detection subroutines take the same form (shown in *figure 15*). They begin by initialising the A register with the coin value to be added to the total. The MVI A, constant instruction, is used for this. Thus, in the case of the penny, the constant is 1, in the case of the 2p coin the constant is 5, and so on until for the £1.00

coin, the constant 64_{16} (hexadecimal equivalent of binary 01100100) is loaded into the A register. The H on the end of the constant instructs the TMS8080A that the number is a hexadecimal number.

Once the constant is in the A register, an OUTPUT instruction clears the appropriate coin detection flip-flop $\overline{Q}$ output back to 1. For example, the pound flip-flop is cleared with an OUT 7 instruction, the penny flip-flop is cleared with an OUT 1 instruction, and so on. Once the coin detection flip-flop has been cleared, a JMP TOTAL instruction causes a jump to the subprogram that adds the constant in the A register to the total in the B to E registers.

Display request

The display request interrupt subprogram (*figure 16*) starts in ROM at location 56. This is more complicated than the other interrupt sequences.

Basically what happens is that the E register contents are written (sent) to output port 11, the D register contents to ouput port 10, the C register contents to output port 9, and the B register contents to output port 8.

Of course, since the contents of the A register are sent out on the data bus during an OUT instruction, the contents of the appropriate B, C, D or E registers must be moved over to the A register just prior to the OUT instruction. For example, to output the contents of the E register, first a MOV A,E is used to get the contents of E over to A. Then, an OUT 11 instruction sends this data out of the two least significant digit LEDs (the pence display).

Once all of the outputs have been sent to the displays, the OUT 13 instruction turns on the display. Next, after a program delay sequence, an OUT 0 instruction clears the display request to avoid an endless display request loop. Then an EI and HLT sequence re-enables the interrupt system and causes the processor to wait for a new interrupt or a RESET.

The program also provides a 15 millisecond delay, starting at the LXI instruction to allow time for the display switch to stop bouncing before the microprocessor releases the interrupt input from the display flip-flop.

16

Label	Instruction	Operand	Comments
DISPLAY	MOV	A,E	Move register E to A
	OUT	11	Output pence digits to display
	MOV	A,D	Move register D to A
	OUT	10	Output two least-significant pound digits to display
	MOV	A,C	Move register C to A
	OUT	9	Output next two most-significant pound digits to display
	MOV	A,B	Move register B to A
	OUT	8	Output most-significant pound digits to display
	OUT	13	Turn on display
DELAY	LXI	B,08F9H	Load BC with hex counter value
DIS	DCR	C	Decrement LS byte of counter
	JNZ	DIS	Jump if not zero
	DCR	B	Decrement MS byte of counter
	JNZ	DIS	Jump if not zero
	OUT	0	Set display request flip-flop
	EI		Enable interrupts
	HLT		Wait for interrupt or reset

17

Label	Instruction	Operand	Comments
TOTAL	ADD	E	Add contents of E to A
	DAA		Decimal correction
	MOV	E,A	Send result to E
	JNC	ENABLE	If no carry, go to enable
	MOV	A,D	Otherwise, add 1 to D with decimal correction
	ADI	1	
	DAA		
	MOV	D,A	
	JNC	ENABLE	IF no carry, go to enable
	MOV	A,C	Otherwise, add 1 to C with decimal correction
	ADI	1	
	DAA		
	MOV	C,A	
	JNC	ENABLE	If no Carry, go to enable
	MOV	A,B	Otherwise, add 1 to B with decimal correction
	ADI	1	
	DAA		
	MOV	B,A	
ENABLE	EI		Enable interrupts
	HLT		Wait for interrupt

Keybounce like this is an unwelcome phenomenon that causes a closing switch to register many short off/on operations when its action is analysed. These can, of course, be easily interpreted as bona fide off/ons by a high speed digital circuit, so some time has to be allowed for the switch to settle down. Otherwise, in this case,

16. The display request interrupt subprogram.

17. Subprogram that computes the total of 'pounds' sorted.

18

```
0060                        ORG     60H
0060  83              TOTAL: ADD     E
0061  27                     DAA
0062  5F                     MOV     E,A
0063  D27B00                 JNC     ENABL
0066  7A                     MOV     A,D
0067  C601                   ADI     1
0069  27                     DAA
006A  57                     MOV     D,A
006B  D27B00                 JNC     ENABL
006E  79                     MOV     A,C
006F  C601                   ADI     1
0071  27                     DAA
0072  4F                     MOV     C,A
0073  D27B00                 JNC     ENABL
0076  78                     MOV     A,B
0077  C601                   ADI     1
0079  27                     DAA
007A  47                     MOV     B,A
007B  FB              ENABL: EI
007C  76                     HLT
```

18. The assembled version of the TOTAL subprogram of *figure 17*.

there would be a number of continuous display requests. As this delay is carried out with software, rather than hardware, it can be easily adjusted if necessary.

Subprogram to maintain a total

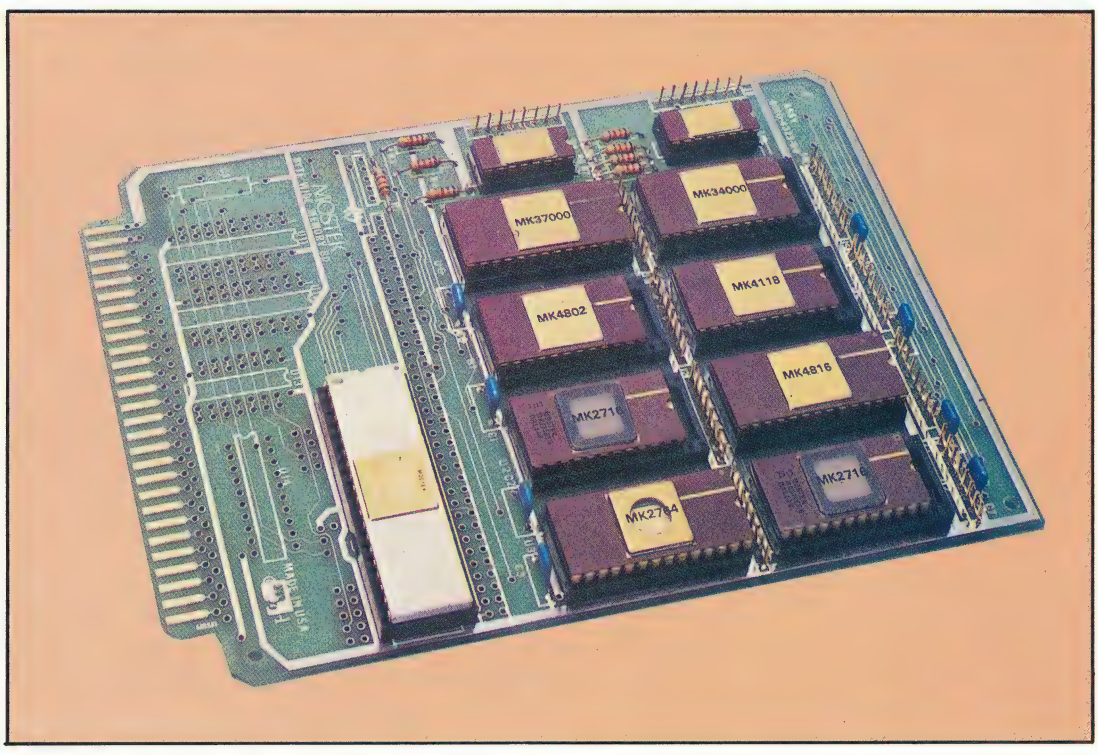
The TOTAL subprogram is shown in *figure 14*. This performs the operation of adding the constant that is loaded into the A register by one of the interrupt subroutines to the current total stored in registers B, C, D and E.

It does this by first adding E to A, with the result stored in A (ADD E). A decimal adjust (DAA) instruction corrects the 24-bit BCD codes and makes the addition decimally correct, at which time the result is stored in E with the MOV E,A instruction.

A conditional jump on no carry (JNC) to ENABLE occurs if there is no carry out from this first byte addition. If there is a carry, 1 must be added to the contents of D by using the MOV A,D to bring D to A, followed by the ADI 1 for the addition to A and the DAA instructions for the necessary correction. The result in A is sent to D with the MOV D,A operation.

Similarly, another jump conditioned on a carry (JNC) is used to see if the program is to jump to ENABLE or add 1 to the contents of C with similar program steps as for register D. If 1 is added to C, the carry must again be checked to see if the program is to jump to ENABLE or add 1 to the contents of B.

Once the complete sum is formed, the program re-enables the interrupt system with the EI instruction at location ENABLE, followed by a HLT to force the processor to wait for the next coin or display interrupt.



Right: computer memory board comprising ROM (MK37000 and MK34000), RAM (4802, 4118, 4816) and EPROM (2716, 2764). (Photo: MOSTEK).

The program requirements for this simple system are now complete. Its features are readily expandable: by adding a sequence of instructions at the end of the DISPLAY subprogram that would cause the total value to be sent out in serial ASCII character form, the system could communicate directly with a bank computer. Of course, a serial data hardware interface would have to be added as an output port 14, but this would present no real problem. If the number of coins counted in each denomination were required, the circuitry to do this could also be added.

Much of the simplicity of both the hardware and the software of this example is due to the 8-bit microprocesor's capability and the relative efficiency of the TMS8080A's instruction set. This is one of the advantages of using an 8-bit processor for such a problem.

The program would, of course, have to be assembled before it can be used by the system. That means that the mnemonic instructions would have to be converted to their respective machine codes, all multiple byte instruction locations included with their data and addresses, and all labels assigned addresses. This would normally be done by computer, but it can be done manually. As an example, *figure 18* shows the assembled version of the TOTAL subprogram.

Summary

So, what have we learnt about 8-bit microprocessors?

1) 8-bit microprocessors offer instruction and interrupt features which are particularly useful for control, communications and numerical computational problems requiring medium speed data transfers and relatively accurate numerical results.

2) Microprocessors with internal registers may provide enough on-chip data storage, so that external random access data memory is not required.

3) The availability of an interrupt system can greatly simplify the exchange of information between the processor and external devices, resulting in simple hardware and programs.

4) Full use should be made of the high performance versatile integrated circuit support devices to simplify both the hardware and software design.

5) As with all microprocessors, the overall design effort in developing an 8-bit system must be defined by a flowchart and then broken down into hardware subsystem designs and the development of their related subprograms.

Glossary

keybounce	rapidly occurring, repeated on/off effect, caused when a switch is closed
on-chip storage	memory positions that are actually held on the microprocessor chip. Usually in the form of registers
optoelectronic sensor	a light sensitive device. When the beam of light striking the sensor is broken, the sensor's output changes and in this way it can be used to provide an on/off (0/1) signal compatible with digital circuitry
pushdown stack	storage locations arranged in the form of a stack. When a data item is sent to the stack it is assigned the top placing. The device that was previously held in this place is pushed down to the one below, and so on throughout the stack. The first item to be retrieved is the last item to be stored
stack	storage locations arranged one on top of the other. Depending on the type of stack, data is either stored and retrieved from the top of the stack, or stored at the top and retrieved from the bottom

17 COMMUNICATIONS

Protocols-2

Packet SwitchStream protocol

British Telecom's Packet SwitchStream data communications network is based on CCITT protocol X.25 and associated recommendations. This X.25 protocol corresponds to the three lowest levels or layers of the open systems interconnection model, and so is often referred to as the **three-level protocol** of X.25 networks.

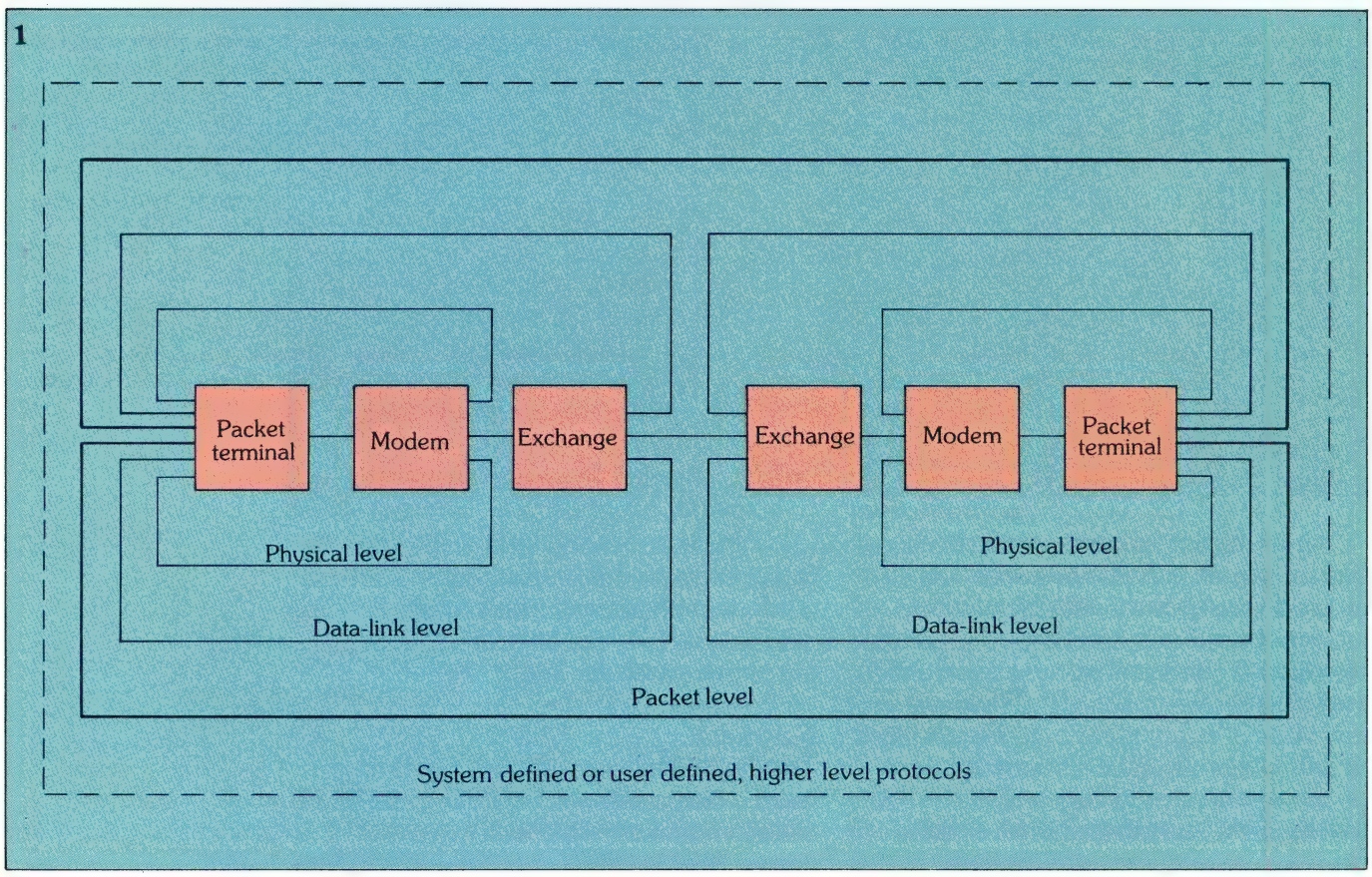
The relationship between these three specified levels of the Packet SwitchStream protocol (or any X.25 protocol, for that matter) is shown in *figure 1*. Note that higher protocol levels are not actually specified by either CCITT or the ISO, although they may be specified by the network authority (British Telecom in the case of Packet SwitchStream). It is possible that in the future, higher level protocols may be defined by CCITT or ISO, however.

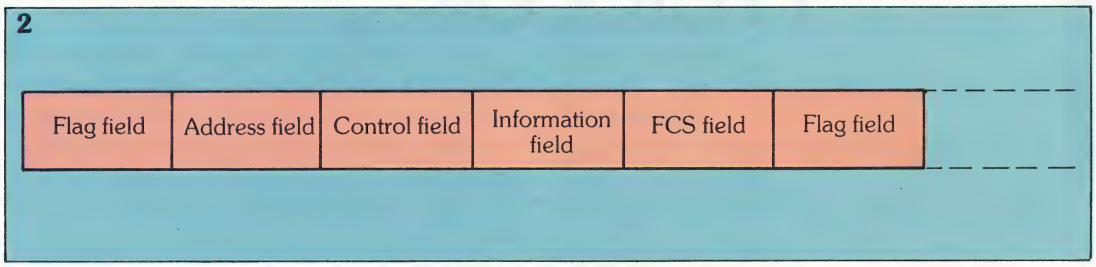
1. The three-level X.25 protocol (physical, data-link and packet levels) and its relationship to the system defined higher levels of protocol.

The physical level

Sometimes referred to as the **bit level** (because the data transmitted at this level is in the form of bits), the protocol level defines: the hardware, i.e. plug and socket, and pin assignment; the electrical signals, i.e. voltages; and the interchange circuits between DTE and DCE.

Recommendation X.21 bis contains the complete specification of this level when connecting synchronous terminals to modems. It is a refinement of



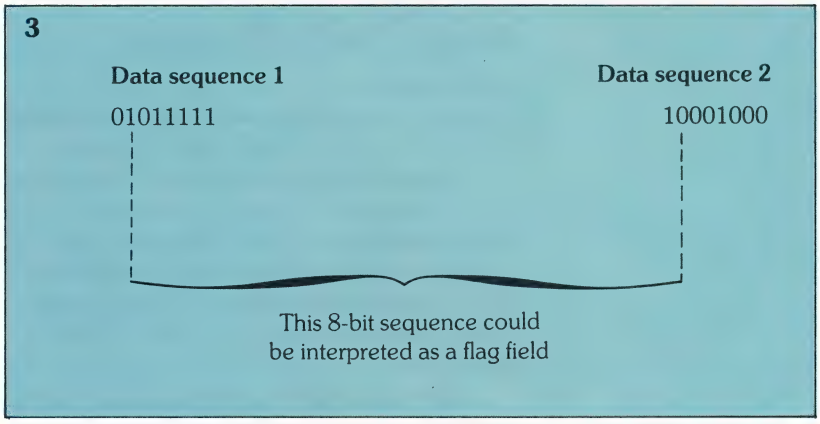


2. An HDLC frame comprising flag field, address field, control field and frame check sequence (FCS) field. The information field may or may not be present depending on the particular frame.

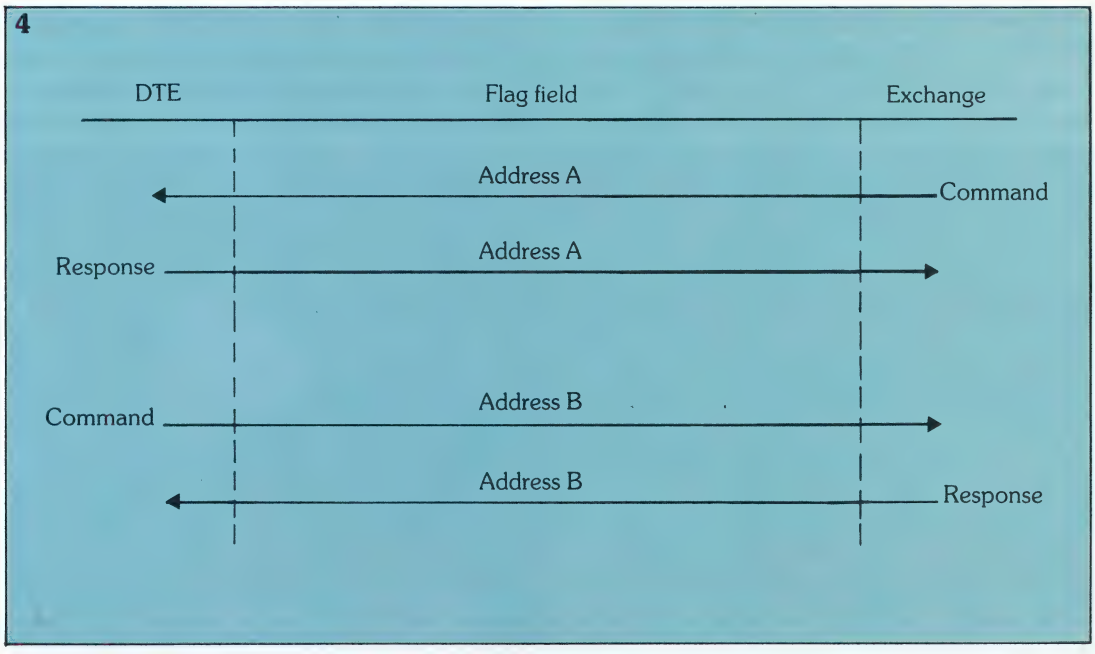
recommendation V.24: the recommendation which specifies connection of asynchronous terminals to modems.

The data-link level

This second level protocol is based on the **high level data-link control** (HDLC) protocol (a standard specified by the ISO) which controls the physical link between the two ends of the physical circuit maintained by level 1 protocol. It detects and corrects data errors during transmissions between DTE and exchange.



3. Bit stuffing overcomes this potential problem by inserting a 0 immediately after any string of five 1s.



4. The address field of the HDLC frame contains an 8-bit code to identify it either as a command frame (address A), or as a response frame (address B).

Information is transmitted in blocks of data or **frames** and so this level is often known as the **frame level**. Each frame comprises user data, protocol information from level 2 and some information from level 3, as shown in *figure 2*. There are, in fact, three different types of frame all based on this format. An HDLC frame contains four components: the **flag field**; the **address field**; the **control field**; and the **frame check sequence (FCS) field**.

The **information field** of the level 2 frame contains data corresponding to level 3 information and the user's message – this field may or may not be present, depending on the particular frame.

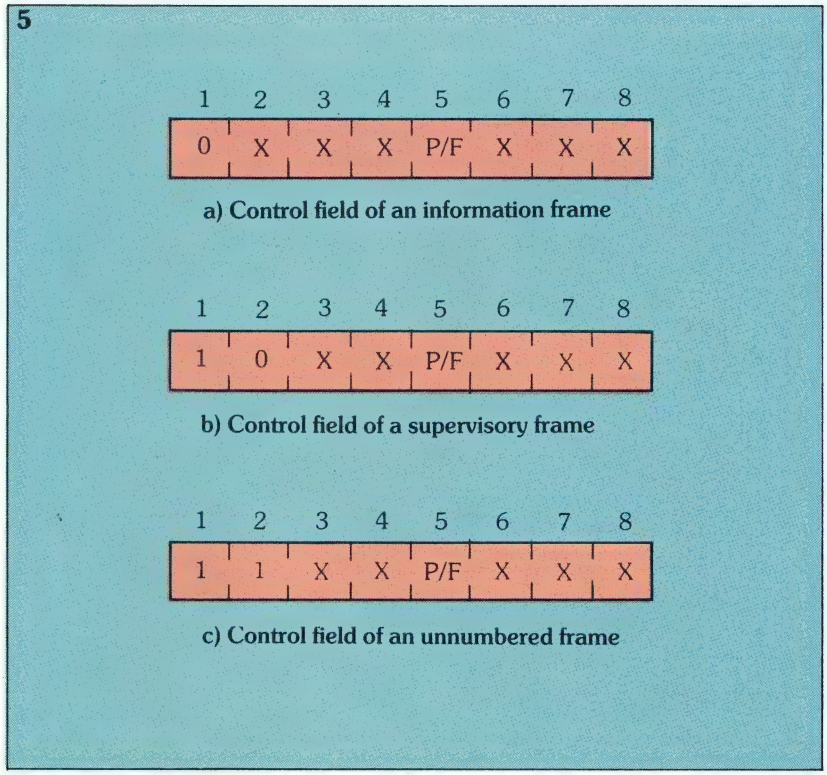
Flag field

The flag field comprises a unique 8-bit word which marks the beginning and end of each frame. A frame which has the correct 8-bit flag field at either end is said to

be **delimited**.

The 8-bit word used in the flag field is not used at any other time by any other field. It is not even allowed to be used *between* data words in the frame. If, say, the flag field is defined by the 8-bit sequence:

01111110



5. The three different types of HDLC frame: (a) information; **(b)** supervisory; and **(c)** unnumbered. It is the control field which differentiates between them.

(the actual sequence used in HDLC) then two consecutive data words cannot hold this sequence. *Figure 3* shows how this may occur. If the correct sequence is incorrectly interpreted as a flag field, then data-link level control is lost.

To overcome this problem, a technique known as **bit stuffing** is used – transmitting DTEs monitor the data between legitimate flag fields. If a sequence of five 1s occurs at any time, a 0 is inserted at the end of the string, regardless of whether the next bit is a 1 or a 0. The remainder of the frame continues unaltered (unless other sequences of five 1s occur). Each sequence of five 1s therefore increases the frame length by one bit.

The receiving DTE similarly monitors the received frame. If any sequence of five 1s is received, the following bit is checked to see if it is a 0 or a 1. If it is a 0 then it is removed.

A 1 instead of a 0 indicates one of two things – either a flag field has been found, or an error has occurred. Should the following bit be a 0, a flag field is confirmed. (This pattern might also occur due to transmission error, but the probability is small.)

Address field

The address field contains an 8-bit code which shows whether the frame has just been originated or if it is an acknowledgement. A recently originated frame is known as a **command frame**, and the address field is given the 8-bit sequence:

00000011

which is known as **address A**.

An acknowledged frame, on the other hand, is known as a **response frame**, and the address field has the sequence:

00000001

which is known as **address B**.

Figure 4 illustrates the way in which the addresses in the address field are used.

Control field

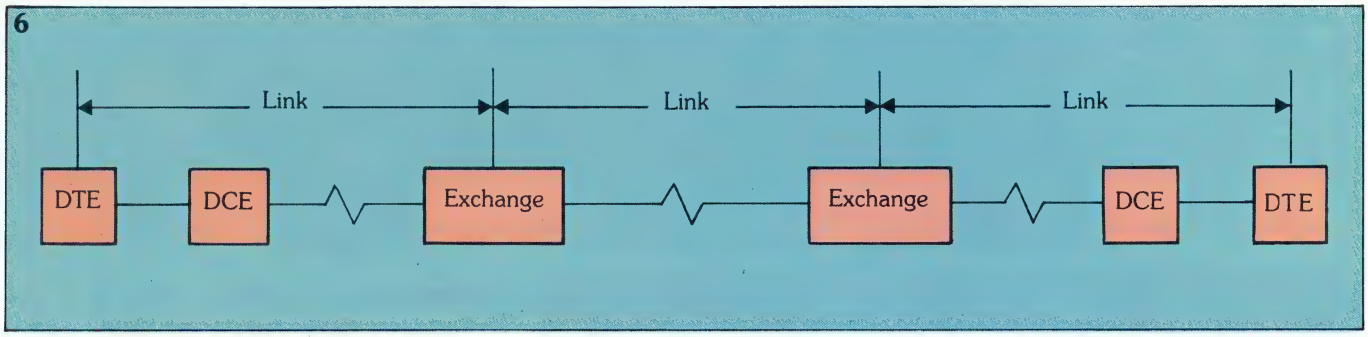
Earlier we said that there are three types of frame using the HDLC format – it is the control field which differentiates between them.

Figure 5a illustrates the format of the control field of an **information frame**, i.e. a frame which carries user information. Bit number 1 in the control field indicates this by presence of a 0. An information frame is the only type of frame which carries a level 3 packet in the information field.

The **sequence number** of the frame is given by bits 2, 3 and 4. This is used by the receiving equipment to check that the frames are in the correct order. Similarly, bits 6, 7 and 8 provide the **acknowledgement number**.

If any frame is transmitted, but not acknowledged, it is retransmitted. So that the receiving equipment of the link can identify an original or a retransmitted frame, bit 5 of the control field – the **poll-final** (P/F) bit – is set to 0 (original) or 1 (retransmitted).

Figure 5b shows the control field of a **supervisory frame**. There are no level 3 packets in a supervisory frame, so a sequence number is not required. An



acknowledgement number is, however, required, as such a frame can be used to acknowledge receipt of an information frame. Bits 1 and 2 (set to 10) indicate that the frame is supervisory, while bits 3 and 4 indicate which type of supervisory frame (**receive ready, receive not ready,** or **reject**) it is.

The control field of the third type of frame – an **unnumbered frame** – is shown in *figure 5c*. This frame's name is derived from the fact that neither sequence number nor acknowledgement number exist. Bits 1 and 2 (set to 11) indicate an unnumbered frame, while bits 3, 4, 6, 7 and 8 indicate which of four types of unnumbered frame it is.

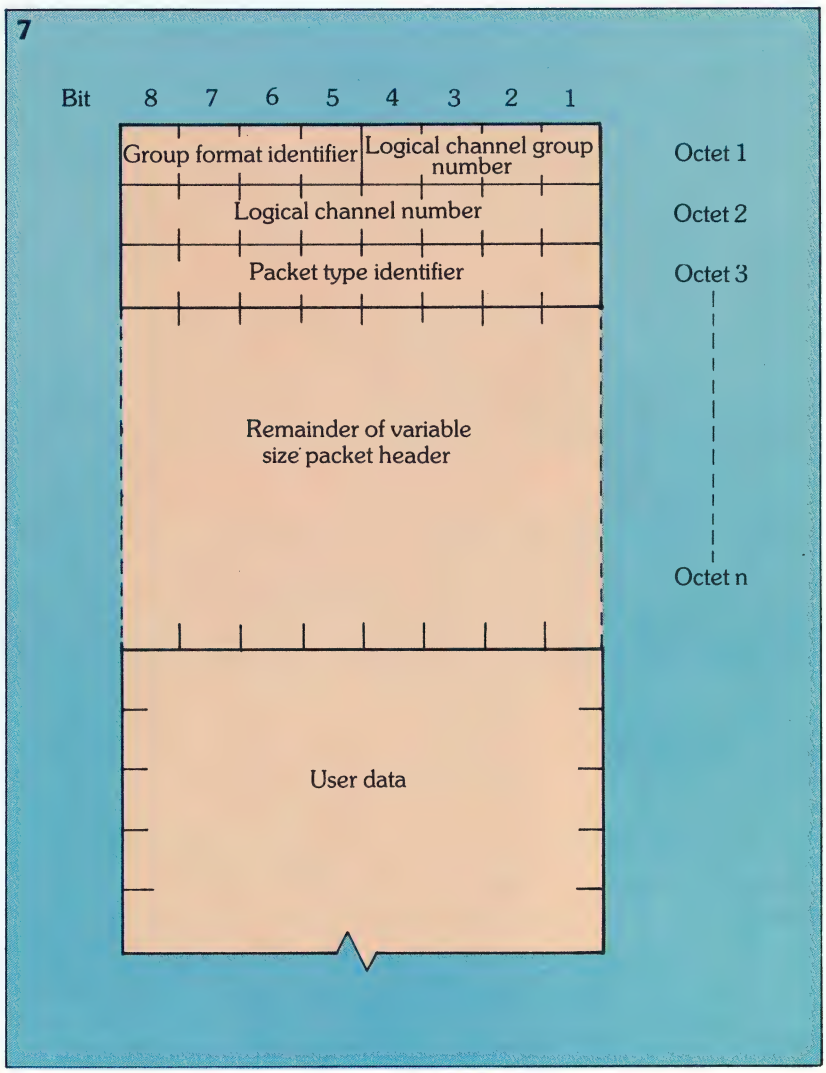
Information field

This field only exists in information frames and contains the user's message in the form of a level 3 packet. Packets will be discussed later in the chapter.

Frame check sequence field

The purpose of the frame check sequence field is purely to check that data in the address, control and information fields is received correctly. The process used is known as a **cyclic redundancy check** and if an error is detected the frame is discarded.

A complex algorithm (based on a **generator polynomial**) the cyclic redundancy check is applied to data at the transmitting equipment and the result is transmitted in the frame check sequence field. The receiving equipment performs a second algorithm on the received data (including the data in the frame check sequence field). If the data has been received without errors, the answer to this second algorithm will *always* be the same – whatever the data. If the answer to the second algorithm is not the same, an error has been detected.



Before we move on to look at level 3 protocols we'll consider some important points:

1) Level 2 protocols exist purely to transmit a level 3 packet across a link.

2) A link is the transmission path between transmitting and receiving equipment. As *figure 6* shows, there may be many links

6. A link is the transmission path between transmitting and receiving equipment.

7. All packets follow this general format – comprising a number of 8-bit bytes or octets.

between two DTEs which are transmitting and receiving data.
3) All of the HDLC information (known as an HDLC header) in the frame, only exists between transmitting and receiving equipment of one link, say DTE 1 and exchange 1 of *figure 6*. At the end of the link, it is removed and a new HDLC header is added, ready for transmission over the next link.

Octet 1 comprises two parts: a 4-bit code, known as the **general format identifier** (GFI); and a second 4-bit code known as the **logical channel group number** (LCGN). The general format identifier allows different formats of packet header to be indicated. A **logical channel** is a communication channel between two DTEs. The term 'logical' is used because all communications are transmitted in a multiplexed way over a single physical link between the transmitting DTE and the exchange. The difference between each communication is therefore controlled by the DTE.

There is a total of eight possible groups of logical channels indicated by the logical channel group number. Each group of channels holds a maximum of 256 channels. Octet 2 of the packet header contains an 8-bit binary number which identifies which of the 256 channels the packet is in. It is easy to see that, with eight groups of channels, over 2000 different logical channel communications may be accessed by a single DTE.

There are 14 different types of packet, identified by octet 3 of the packet header, the **packet type identifier**. *Table 1* lists the 14 packet types and corresponding

Table 1
Packet types and identifier codes

Packet type		Octet 3
From terminal to exchange	From exchange to terminal	8 7 6 5 4 3 2 1
Call set up and clearing		
Call request	Incoming call	0 0 0 0 1 0 1 1
Call accepted	Call connected	0 0 0 0 1 1 1 1
Clear request	Clear indication	0 0 0 1 0 0 1 1
Terminal clear confirmation	Exchange clear confirmation	0 0 0 1 0 1 1 1
Data and interrupt		
Terminal data	Exchange data	X X X X X X X 0
Terminal interrupt	Exchange interrupt	0 0 1 0 0 0 1 1
Terminal interrupt confirmation	Exchange interrupt confirmation	0 0 1 0 0 1 1 1
Flow control and reset		
Terminal rr	Exchange rr	X X X 0 0 0 0 1
Terminal rnr	Exchange rnr	X X X 0 0 1 0 1
Terminal rej		X X X 0 1 0 0 1
Reset request	Reset indication	0 0 0 1 1 0 1 1
Terminal reset confirmation	Exchange reset confirmation	0 0 0 1 1 1 1 1
Restart		
Restart request	Restart indication	1 1 1 1 1 0 1 1
Terminal restart confirmation	Exchange restart confirmation	1 1 1 1 1 1 1 1

Because of these features, HDLC protocol is said to have **link significance**. The level 3 network protocol is known as a **packet level protocol** and it has **end-to-end significance**, i.e. the packets it carries are transmitted intact from one DTE to another.

The network level

In the same way that user information is transmitted at the data-link level as a combination of packet plus HDLC header, at the network level, a user's message is transmitted with a **packet header**.

Figure 7 shows the format of all packets. Although each bit is transmitted serially, it is easier to consider the packet as being a number of **octets** (i.e. 8-bit bytes) in a vertical representation as we have shown.

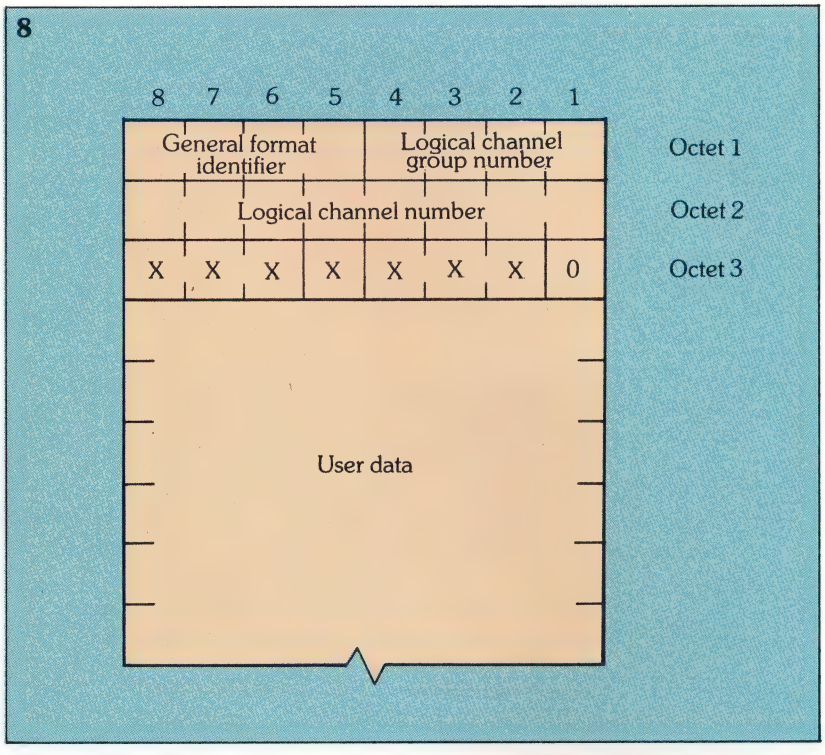
identifier codes. As you can see, the type of packet depends on whether it comes from DTE originating the call, or DTE receiving the call. The remaining octets of the variable size header depend on the packet type.

In a call request packet, for example, octet 4 is known as an **address length field**. This indicates the number of numbers in the called and calling DTEs' network user addresses. As we know from *Communications 15*, a DTE's network user address is generally of 12 numbers. However, if use is made of the two optional sub-addressing digits, the network user address increases to 14 numbers. Octet 4 is divided into two 4-bit codes and the length of the called terminal's network user address is represented in binary by bits 0 to 3, while the length of the calling terminal's network user address is represented by bits 4 to 8.

The octets following octet 4 of a call request packet (the **address field**) contain BCD words corresponding to the decimal numbers of, first the called terminal's address and, second, the address of the calling terminal.

Once the call has been set up between DTEs, the network user addresses are not transmitted in further packets. The logical channel details given by octets 1 and 2 provide sufficient information to ensure that each subsequent packet arrives at its destination. Packet headers of the following packets therefore only require details of the logical channel and some sort of counting procedure so that packets are received in the same order that they were transmitted. A similar count method to that in control fields of HDLC level 2 information frames is used, as shown in *figure 8*.

Only three octets are used in the headers of such **data packets**. The first two are as previously discussed. Octet 3 (the packet type identifier), however, now contains a **send tally** number (bits, 2, 3 and 4) and a **receive tally** number (bits 6, 7 and 8). Bit 5 is known as the **more data bit** which is set to 1 if more data is to follow, but is 0 if this packet is the last in a sequence. Bit 1 is always 0 in a data packet.



8. A data packet header comprising three octets. Bits 2, 3 and 4 of octet 3 contain the send tally number, while bits 6, 7 and 8 contain the receive tally number.

Extra facilities

After the address field of call request packets, optional fields exist which allow access to a number of facilities. Octet 4, for example, is the **facility length field** which gives, in binary, the number of octets in the following **facility field**. If the facility length field is 0, then no facilities have been requested and so no facility field is included.

When present, the facility field contains codes for choosing extra facilities. These include:

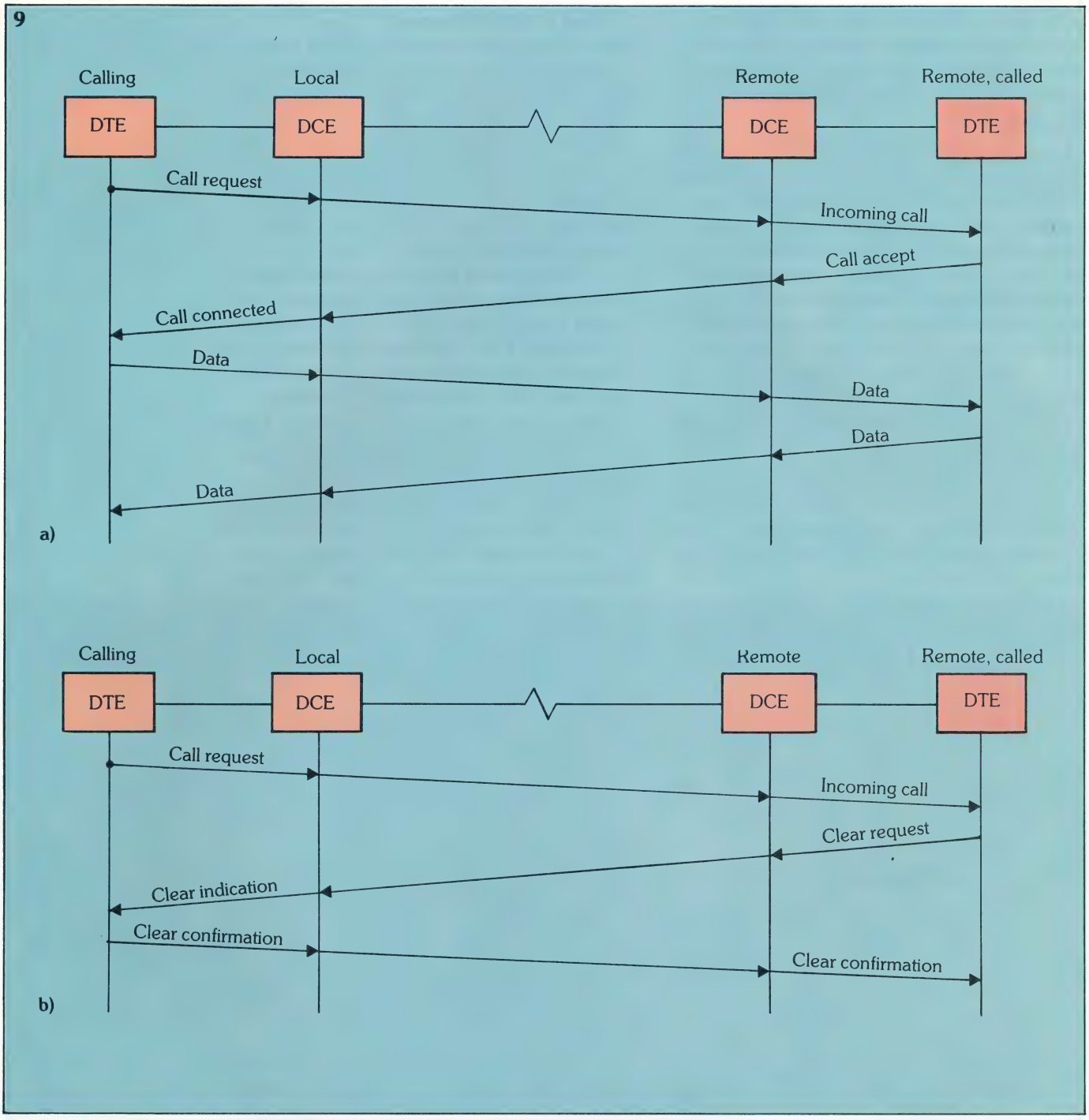
1) **Transfer charging** – where the called terminal's user is billed for the charge of the call.

2) **Packet size selections** – normally user data is in batches of 128 octets; this facility allows batches of 256, 512 or 1024 octets.

3) **Closed user group** – access to a select number of terminals, acting like a network within the public network, allowing a higher degree of privacy and security than available on the public network.

4) **Fast select** – a facility which allows a terminal to send a single packet and accept one in return. Such a **minicall** is useful when only a small amount of data (up to 128 octets) is to be transmitted.

5) **Call duration** – the exchange will send back a packet indicating the length of the call. This is useful if a user requires an estimate of the amount to be charged, before the bill arrives.



9. (a) Call accept; and (b) call refusal procedures for establishing a call between two DTEs.

Establishing a call

The procedures followed by DTE and DCE if a call is to be accepted are shown in *figure 9a*; call refusal procedures are shown in *figure 9b*. To establish a call, the transmitting DTE sends a call request packet to the local DCE. The remote DCE sends an incoming call packet to the remote DTE to indicate that a call has been requested. This is, in fact, the same packet, known simply by another name at each end of the logical channel.

The called DTE inspects the incoming call packet header and decides whether or not to accept the call. If the call is accepted, a call accept packet is sent to the DCE; if the call is rejected, a clear request packet is sent. These are sent to the calling DTE as a call connected packet or a clear indication packet, respectively.

If the call is accepted, data packets are now exchanged between DTEs, but if the call is refused the calling DTE sends a further packet – a clear confirmation – to the called DTE.

The logical channel chosen by a DTE for an outgoing call is the lowest available number. The remote DCE decides the logical channel number for an incoming call. In this case the highest available number is used. A call is only established when a calling DTE receives a call accepted packet having the same logical channel number as the call request packet.

Call disconnection entails a DTE sending a clear request packet, and the remote DCE sends the corresponding clear indication packet to the remote DTE. The remote DTE responds with a terminal clear confirmation packet when all outstanding data packets have been sent. This is returned to the local DTE as an exchange clear confirmation.

Packet SwitchStream

One of the advantages of an X.25 level 3 protocol is that it need only be applied at the DTE/DCE (i.e. terminal/exchange) interface. The network itself (i.e. between exchanges) is not restricted to X.25 implementation. Network users would not even be aware of this. Packet SwitchStream, however, *does* use an X.25 level 3 protocol within the network.

Higher level protocols, especially those of transport and session layers, are under consideration by CCITT and other authorities. Their eventual implementation will mean that different types of networks (not just X.25) may be used simultaneously by one terminal, and different types of terminal will be able to communicate. The eventual effect of the higher level protocols may even be to allow all types of information, e.g. speech, television or data, to be transmitted globally on one network.

Glossary

bit level	synonym for the physical layer of the open system interconnection model
bit stuffing	technique used at the link level in HDLC, in which the transmitting DTE monitors the data stream and inserts an extra bit if a certain sequence is detected
cyclic redundancy check	HDLC error detection procedure in which algorithms are applied to transmitted and received data
frames	a section of data transmitted using HDLC level 2 protocols over a link in the Packet SwitchStream data network
high level data link control (HDLC)	the protocol in the data-link layer of the Packet SwitchStream data network
logical channel	a communications channel over which packets are transmitted. Many logical channels may exist over one physical connection
minicall	transmission of one packet from a calling DTE to a called DTE, and reception of an acknowledgement packet in return
packet header	data added to user information which determines the destination of the packet and other facilities
packet level	level three, the network layer of the Packet SwitchStream data network

ELECTRICAL TECHNOLOGY

Pulses on transmission lines

In the previous *Basic Theory Refresher* we found that the phase shift between voltages and currents at two points on a transmission line, one metre apart, is given by:

$$\tan \phi = 2\pi f C R_o$$

Since, for small angles, $\tan \phi = \phi$, then for a very short length of line, Δl, the phase shift, $\Delta \phi$, is given by:

$$\Delta \phi = 2\pi f C R_o \Delta l$$

This expression may also be written in the form:

$$\Delta \phi = \frac{2\pi f}{u} \Delta l$$

From this we can see that the phase shift between the voltages (or currents) at any two points along a line is *directly proportional to the frequency of the applied signal.*

A voltage generator, transmission line and termination is shown in *figure 1a*. The generator supplies a sine wave, of frequency f, to the line and the line termination is the line's characteristic impedance, R_o. The sine wave supplied by the generator is shown as the black curve in *figure 1b*.

At some point, X, at a distance x from the generator, the voltage is as shown by the red curve in *figure 1b*. The phase lag at X is given by:

$$\phi = \frac{2\pi f x}{u}$$

Now since the phase changes by 2π in a time equal to one full period, T, the time delay of the voltage at x must be given by:

$$\phi = \frac{\tau}{2\pi} T$$

and, as $T = 1/f$:

$$\phi = \frac{\tau}{2\pi f}$$

$$= \frac{x}{u}$$

Now this is a fascinating result since it tells us that the time delay between waves at two points on a lossless, correctly terminated, transmission line is proportional to the distance between the points, but *completely independent of the frequency of the sine wave.*

see page 1492

1. (a) Voltage generator transmission line and termination; **(b)** sine wave representations of voltage at the generator and point X on the transmission line; **(c)** possible shape of a transmitted pulse.

Transmission of a pulse

From our study of Fourier series and spectra, we know that any voltage may be broken up into a number of sine waves of different frequencies. Using this fact, we can re-examine the transmission line of *figure 1a* and find out what happens when a pulse, such as that shown in black in *figure 1c*, is applied.

If we take a theoretical look at the voltage at point X on the line, we might expect to find that each of the sine and cosine waves comprising the pulse occurs there, and that the amplitude of each component at X is identical to the component amplitude at the generator. Also, each component of the waveform is delayed by exactly the same period of time, τ. In other words, all the components of the original pulse should occur at the point X with the same magnitude and relative phase as at the generator and so they will add together to give a pulse identical with that at the generator. This theoretical pulse is shown in red in *figure 1c*.

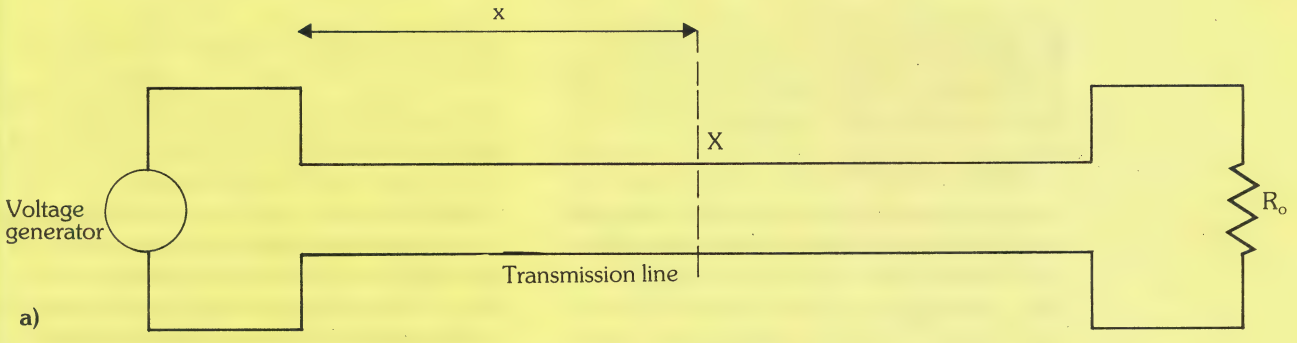
Now this argument will apply to any type of waveform and may be generalised to say that any lossless line, terminated in its characteristic impedance transmits a voltage from one end to the other without causing any distortion of the wave.

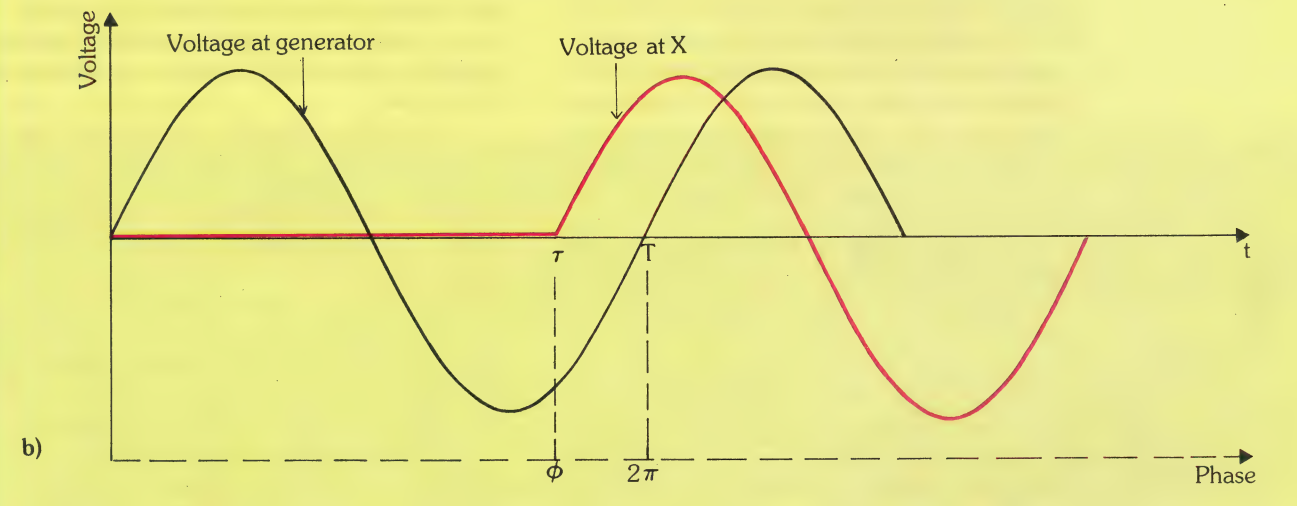
In practise, of course, transmission lines are not lossless: certain losses occur in the series resistance of the wires and, in some situations, in the shunt losses in the dielectric. Obviously, we can't expect a *real* transmitted pulse to be identical to the generated pulse. A possible shape of a pulse is shown by the broken red curve in *figure 1c*.

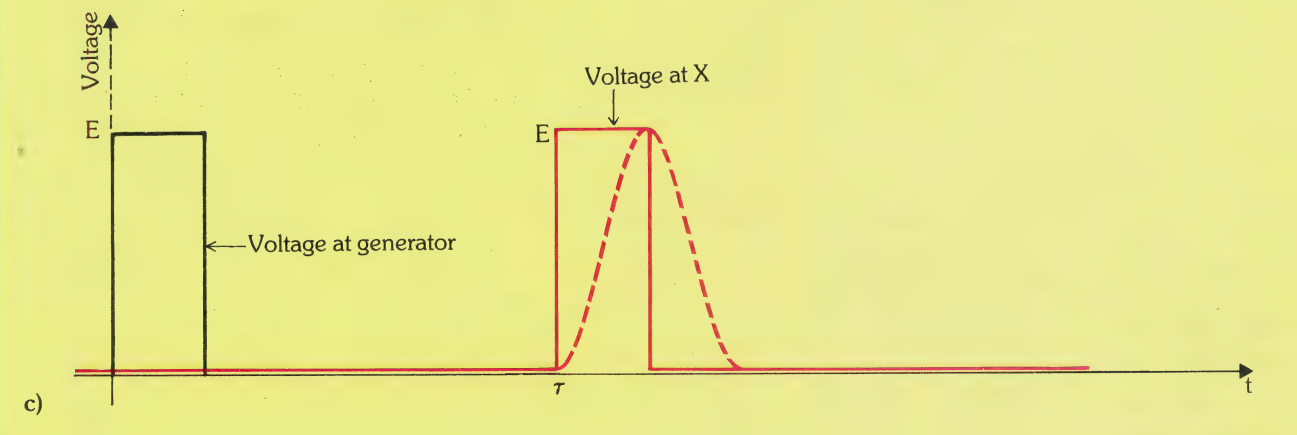
Typically, the longer the transmission line, the broader and flatter the pulse.

If the transmission line is sufficiently long the pulse becomes so distorted that it is impossible to detect its original shape. For this reason, networks which use long transmission lines to transmit pulses must incorporate **regenerators**, spaced at regular intervals, to reconstitute pulses to their original shapes. □

1


x
X
Voltage generator
Transmission line
R_o
a)


Voltage
Voltage at generator
Voltage at X
τ
T
t
b)
ϕ
2π
Phase


Voltage
E
Voltage at X
E
Voltage at generator
τ
t
c)

15

COMPUTERS & SOCIETY

Information technology

What is information

In previous articles in this *Computers & Society* series, we have looked at the various ways in which the new microprocessor-based technologies have been changing the world that we live in. This has encompassed the home, the workplace, our health and education and the need for legislation to provide individuals with protection against the misuse of information. We have also considered how the different worlds of manufacturing, publishing, finance and commerce are being affected, and we have taken a brief look at the more esoteric subjects of artificial intelligence and computer graphics.

Much of what has been discussed involves different mechanisms for gathering, upgrading and distributing information, and therefore comes under the 'umbrella' title of **information technology** or IT. But what actually is IT and how has it come about? In this concluding article, we shall attempt to find out. First, however, we need to define the word 'information'.

The Oxford Concise English Dictionary defines information as: 'informing, telling; things told, knowledge, news'. Chambers Twentieth Century Dictionary gives it as being: 'intelligence given, knowledge'.

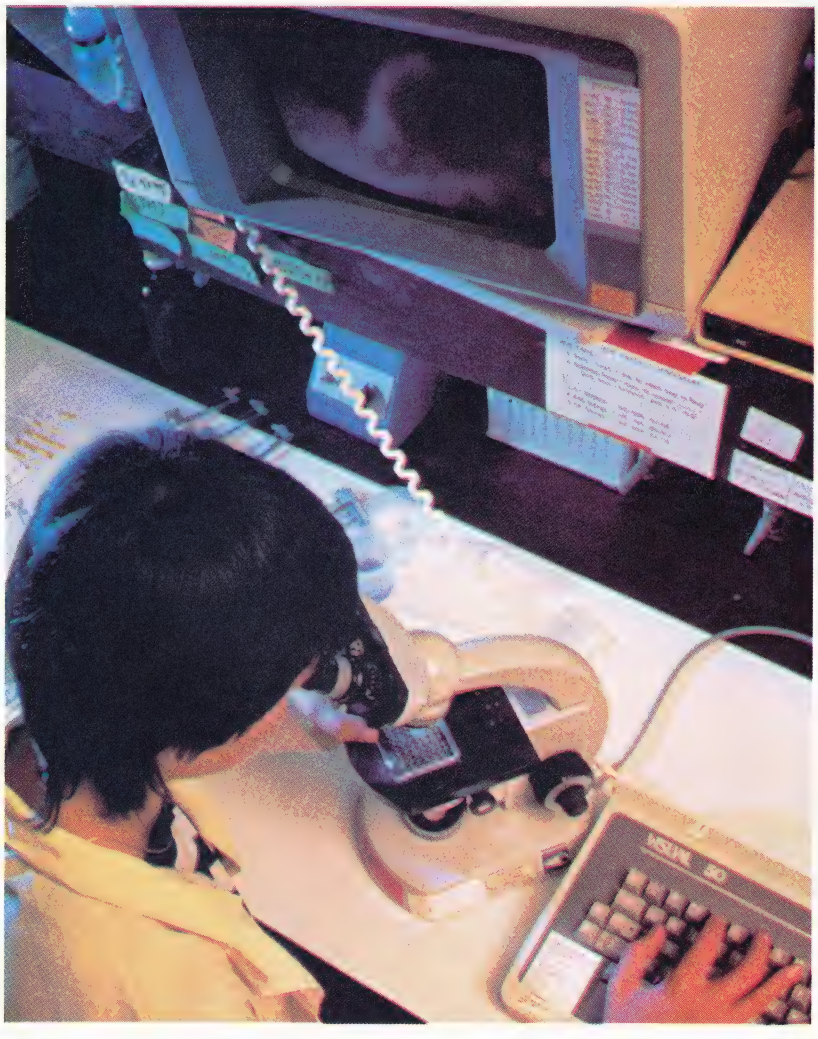
The difficulty presented by this word when applying it to practical matters is that it is intangible. We only become aware of information through its subjective effects. We *derive* information from our observations of our surroundings, from facts or data; we then *convey* this information via various means of communication.

Because of this, it has been necessary to *describe* it in various ways: is it reliable and accurate? is it up-to-date?; can it be verified?; is it complete?; is it relevant and timely?; is it exclusive?; is it predictive? Answers to these questions will determine its practical *value*. This will be of interest to both the sender and receiver of the information and will determine whether or not it is dumped, stored and/or upgraded.

Apart from examining the *content* of the information, telecommunications and software engineers are interested in the symbols, signs and signals that *represent* the information to be communicated.

Thinking of information or knowledge in this way will help us understand the term information technology.

Below: computers are being increasingly used in health care administration. This one stores data about organ donors, and checks for compatibility with suitable patients for transplants.



Science Photo Library/Hank Morgan

What is IT?

Information technology (or 'informatique' in French, or 'informatika' in Russian) has its origins in the technologies relating to the objective, representational aspect of information: its generation, manipulation and distribution. For example, telecommunications, computer engineering, data processing and the office machinery industries.

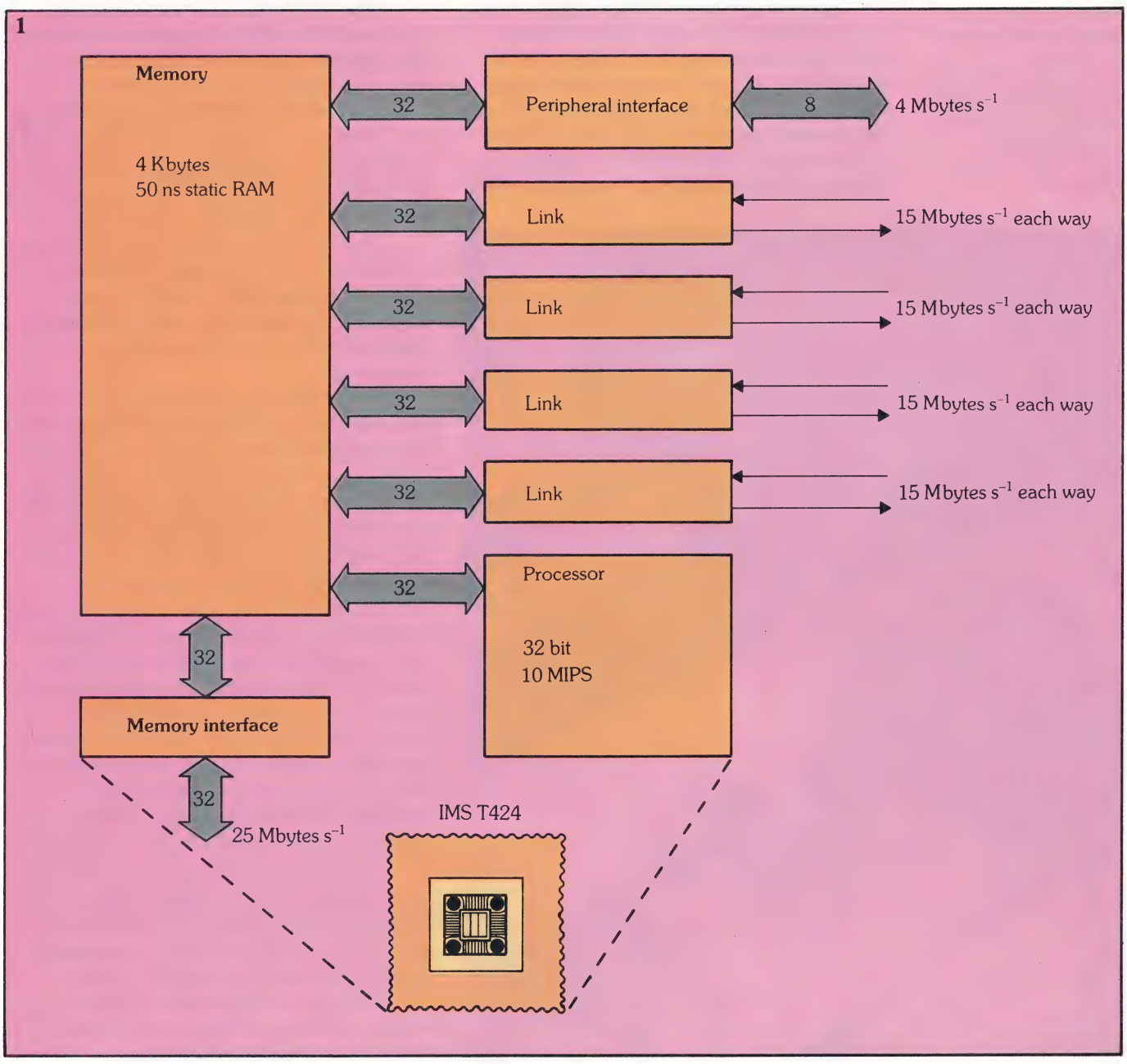
In fact, the products of these industries still form the bulk of information technology products today.

Recently, however, the emphasis has moved away from this data management or '**telematics**' approach towards a greater interest in the range of possible applications. This trend is also concerned with the human or subjective aspects of information (often termed **informatics**): its content, value etc.

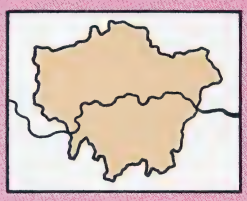
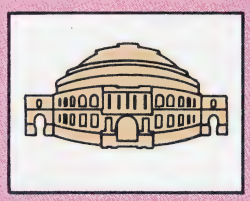
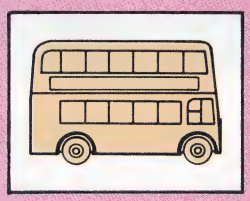
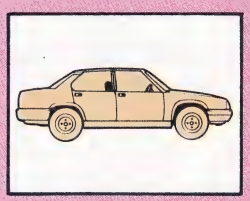
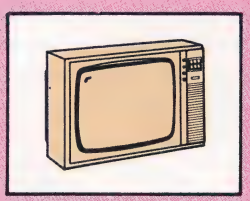
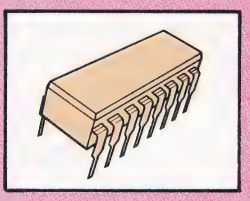
This wider view has led to the following definition of IT from the British Advisory Council for Applied Research and Development (Report on Information Technology, H.M. Stationery Office, 1980).

1. Internal structure of the IMS T424 transputer IC.

2. Forty years ago, the computing power that can now fit into the transputer IC would have needed a machine the size of Greater London! *(After: The Mighty Micro, Dr Christopher Evans).*



2

Year	Technology	Relative size of equivalent computer	
1945	Valves		Greater London
1955	Transistors		Albert Hall
1965	Integrated circuits		London bus
1970	Small-scale integration		Motor car
1975	Medium-scale integration		TV set
1980	Large-scale integration		Brain
1985	Very large-scale integration		Integrated circuit

The scientific, technological and engineering disciplines and the management techniques used in information handling and processing; their applications; computers and their interaction with men and machines; and associated social, economic and cultural matters.

Armed with this definition, we'll now go on to examine some of the contributing factors towards the development and growth of IT.

Miniaturisation

The development of integrated circuit techniques has led to a remarkable increase in the complexity of functions which can be performed by electronic circuits. Although transistors are the only active devices available at present, the range of technologies used in their construction has enabled an incredibly wide variety of circuit to be manufactured.

In 1965, Dr Gordon Moore, President of Fairchild Semiconductors, predicted that the complexity of integrated circuits would double, approximately every year, while their size would remain the same. This rate of miniaturisation has indeed happened, and will probably continue.

Where only fifteen years ago manufacturers were striving to produce ICs with only a handful of internal transistors, today's latest IC has over 250,000 transistors on its silicon chip. The IMS T424 **transputer**, due to begin production shortly, is designed and will be manufactured by the British company INMOS. The term transputer is derived from the words '**trans**istor' and 'com**puter**', since the IMS T424 IC is both a complete computer and made of silicon transistors. Its internal structure is shown in *figure 1*. (See also *Microprocessors 15.)*

The absolute accuracy of the processes required to construct the chips used in such an IC is phenomenal. Individual transistors on the chip are measured in micrometers – it's difficult to comprehend that over a quarter of a million transistors will sit on a chip measuring only 45 mm^2.

In his book *The Mighty Micro*, Dr Christopher Evans compares the size of modern IC computers with the size they would have to be if built from older

Science Photo Library/A Hart-Davis

Left: computers are now an integral part of video editing facilities.

technologies, such as valves and transistors. We can extend this idea to include the transputer: a valve computer of equal power to the transputer IC would have to be about the size of Greater London! (*figure 2*). This level of miniaturisation is a remarkable achievement in a period of about forty years.

The reduction in size is not the only advantage of miniaturisation. The valve computer of *figure 2* would require power of about 500 kW to heat the valves. Remarkably, the transputer dissipates less than 0.9 W, and should run quite happily from a battery pack!

Of course, along with reductions in size comes the advantage of large scale mass production techniques. This has meant that as the number of components able to be fabricated onto a single IC has increased, the cost per component has correspondingly decreased – so our transputer is also only a fraction of the cost of the valve computer. This trend is illustrated by the graph in *figure 3*.

Considerable advances have also been made in **data storage technology**. The 64 kbit solid state RAM chips currently available are soon to be superseded by 256 kbit chips, capable of storing 262,144 bits of data on a 5 mm × 9 mm chip.

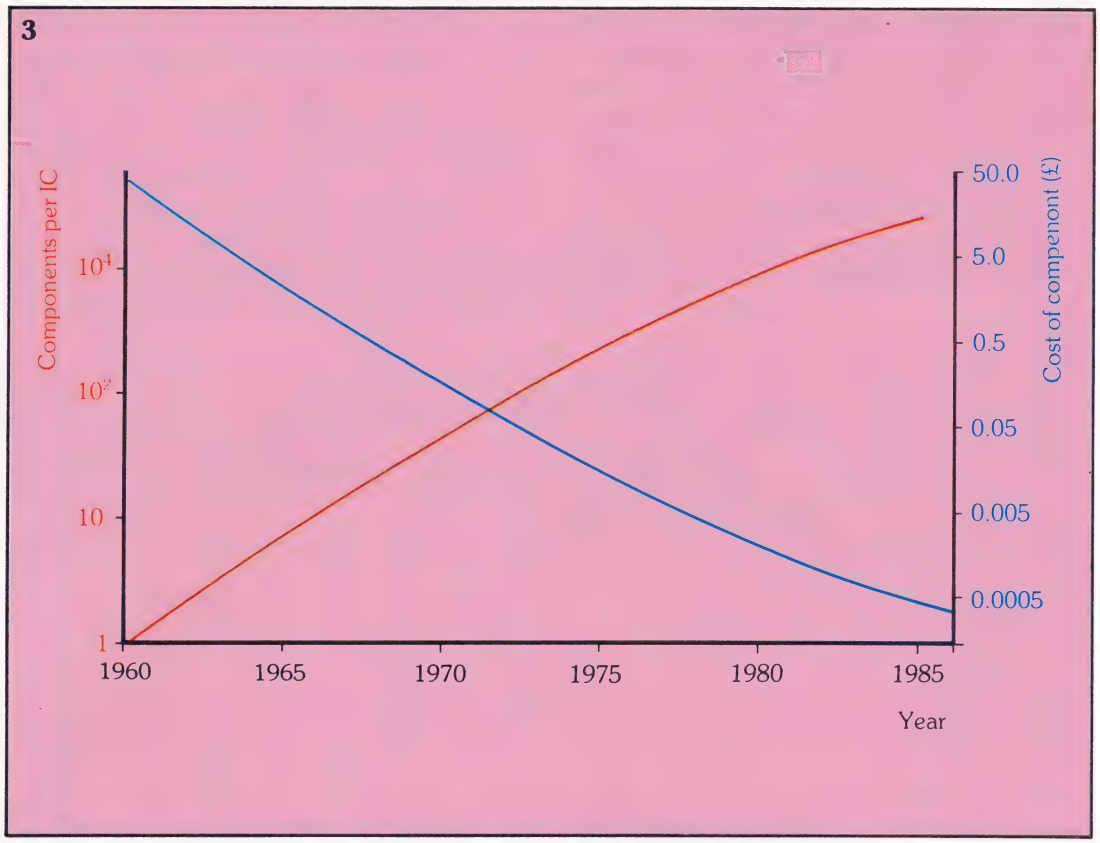
A new high density storage medium, with the trade name 'Drexon', was announced in 1981. A credit card sized piece of plastic, coated with this new organic film, holds about 40 million bits of data! The projected data storage densities of various media are shown in *figure 4*.

A major boost to microelectronic miniaturisation occurred during the American 'race to the moon'. In an effort to reduce on-board weight, yet increase computational power, the microprocessor itself was developed in 1971. Undoubtedly commercial forces would have succeeded in doing this, but backing from government and the military brought this date forward, possibly by a number of years.

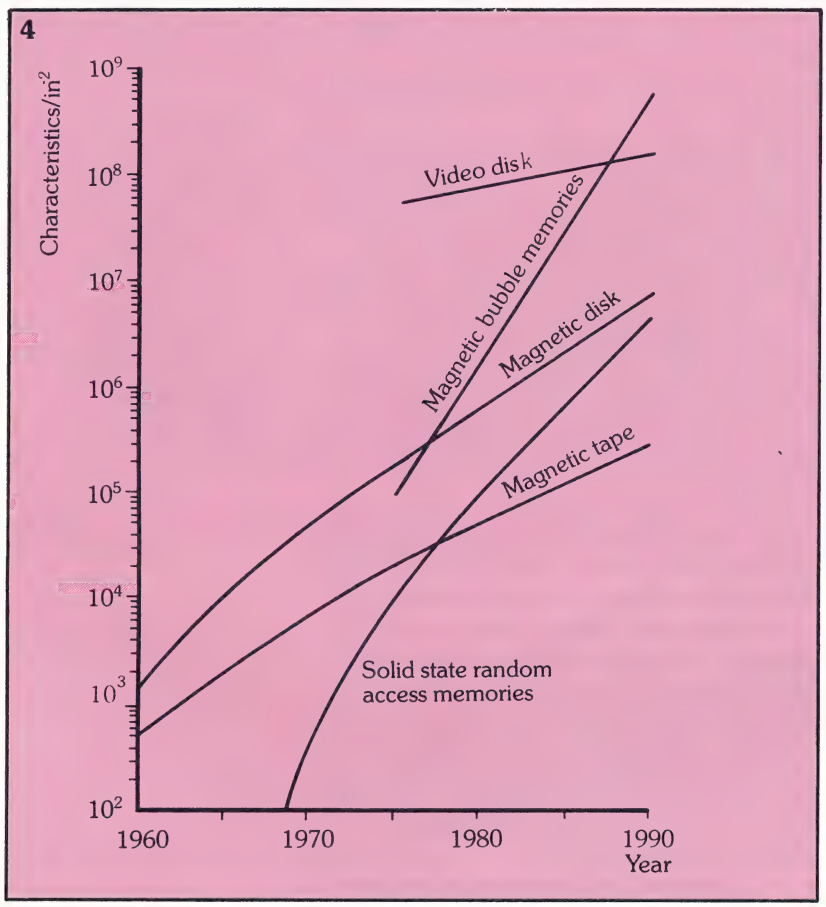
Reliability

Miniaturisation and the development of modular plug-in systems has dramatically improved the reliability of electronic equipment. The mean time between failure, or MTBF, for valve equipment was often measured only in a few hours. Transistors improved this enormously: twenty years ago, MTBFs for single solid state devices such as transistors were in the order of a million hours. If, say, a thousand semiconductors were used in a system, that system's MTBF could therefore be approximated to one thousand hours. For modern semiconductors, MTBFs in the order of 10^9 hours are common and, as only ten or so semiconductors may form a complete

3. As the number of components able to be fabricated onto a single IC has increased, so their individual cost has declined.



4. Improvements in the densities of various data storage media projected to 1990. *(Source: Information Technology – An Introduction, Peter Zorkoczy, 1982).*

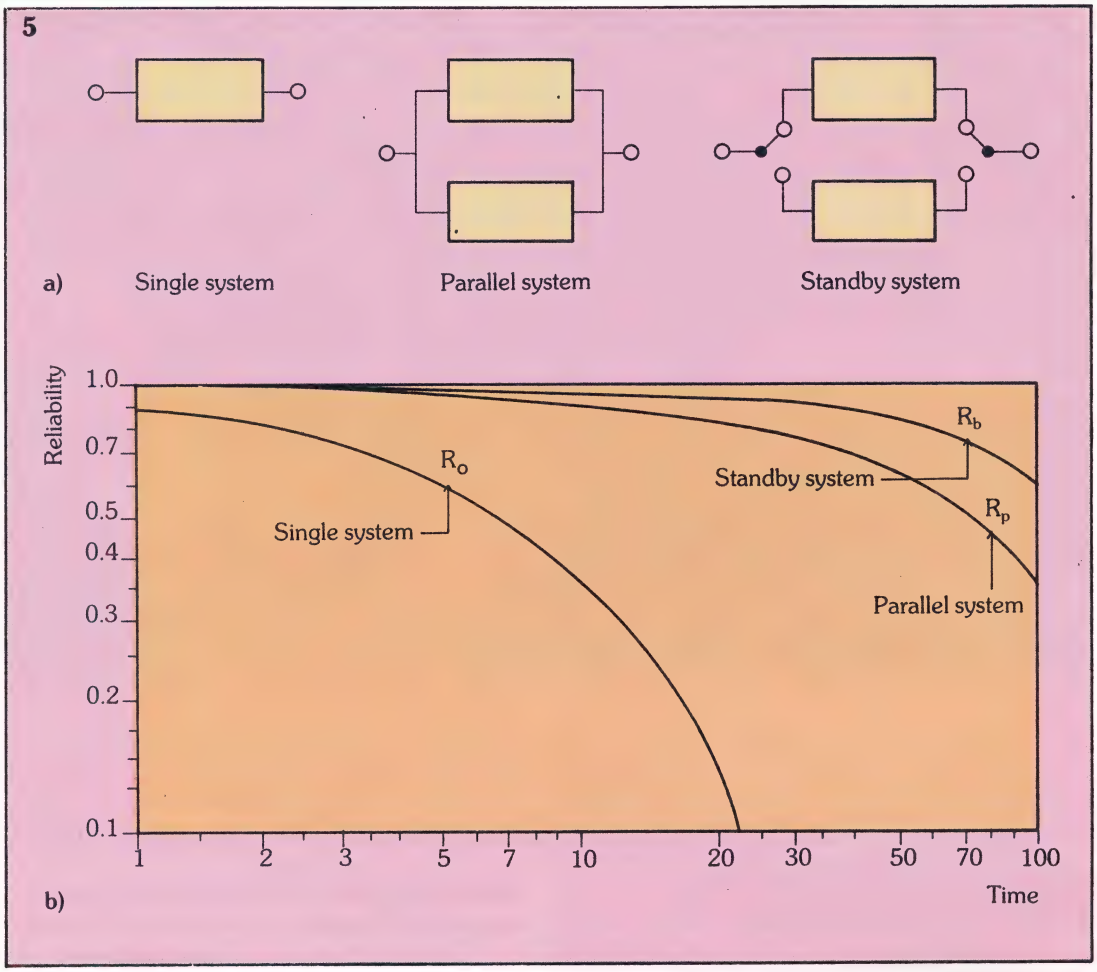


system, the system's MTBF is therefore around 10^8 hours!

The reliability of a single system may also be artificially improved by maintaining a second system in **parallel** with it. This means that if either system malfunctions, the other continues to operate and no running time is lost. If this second system is a **standby system** (i.e. not actually in operation), then further improvements can be made – when the first system breaks down, the standby system is switched into circuit. This is illustrated in *figure 5*. The use of parallel and standby systems only became feasible with the reductions in size, cost and power requirements of IC-based computers.

The influence of the microprocessor

The first large and expensive computers were mainly used for data processing tasks – in banks and research institutions. However, miniaturisation and improvements in reliability, as we have just seen, reduced costs considerably. This naturally led to a wider consumer market for new smaller and cheaper microprocessor-based products. Manufacturers were quick to



5. The use of standby and parallel systems considerably improves system reliability.

recognise this new opportunity and a plethora of useful tools and not so useful gadgets has been the result.

One interesting feature of these new products is the way in which a new innovation is adapted, perhaps with slight modifications, to a wide variety of uses. In *table 1*, some of the products that we have discussed in previous *Computers & Society* articles, such as electronic mail, expert systems, information systems, viewdata, teletext etc. are listed, together with the applications areas in which they can be found in one form or another.

The importance of telecommunications

Telecommunications forms the basis for information technology. Microprocessor-based products of increasing complexity may be developed, but it is their ability to *communicate* which has fuelled the IT revolution.

Until only a few years ago, data or digital communications presented a problem. The only communications network of any size in operation was the telephone network – which is, of course, analogue. Techniques, for example **modems**, for interfacing digitally based computers with the analogue PSTN (public switched telephone network) therefore had to be developed. Also, because of the high data rates needed for computer communications, various methods of fully utilising the available transmission media had to be adopted.

One method of more effectively exploiting transmission links is by accommodating more than one signal in the available bandwidth. This is known as **multiplexing** – a wideband transmission link is shared between many narrower bandwidth requirements. The operation of different types of modems and a detailed discussion of all aspects of data transmission over the PSTN can be found in *Communications 12-17*.

A second method of sharing out

capacity is to only allocate channels to users as they are required (i.e. the link is only temporary). This is achieved by means of **switching**. A comparison of different switching systems can be found in *Communications 15*.

Switching centres have undergone dramatic changes – developing from the manual analogue exchanges to the latest fully automatic computer controlled digital switching centres. At present, those digital systems that are in operation have to cater for both analogue and digital signals. An example of this kind of dual purpose switching centre is BT's System X. The larger exchanges on this system can handle up to 500,000 call attempts per hour.

Table 1
Information technology products and systems and their various applications

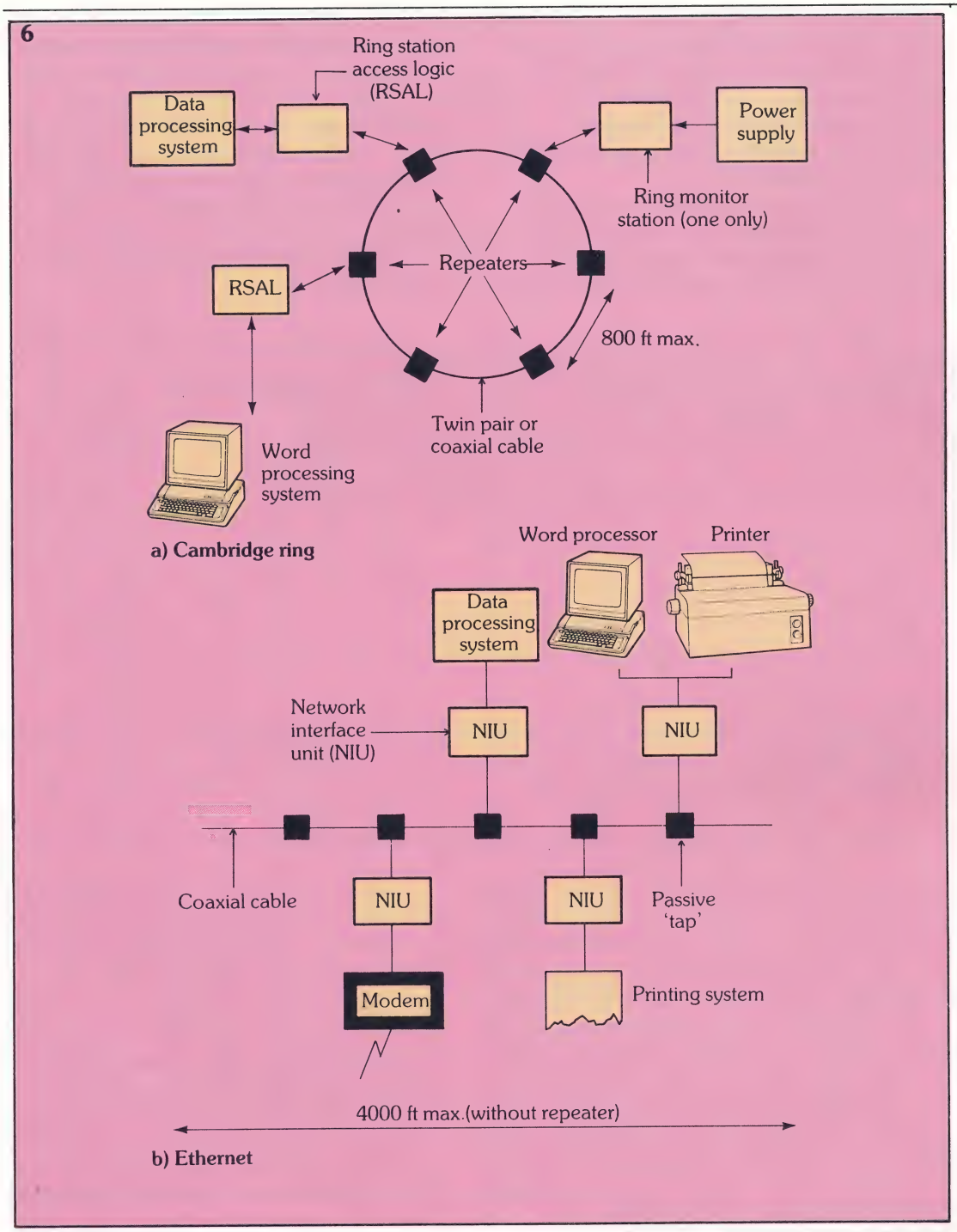
I.T. product or system	Office	Industry	Commerce and finance	Communi-cations services	Health care	Education	Home
Computers	X	X	X	X	X	X	X
Input devices	X		X		X		
Output devices	X	X	X		X		
Storage	X						
Software		X	X			X	X
Computer vision	X	X	X				
Data networks	X	X	X	X			X
Local networks	X	X					X
Long distance networks			X	X	X	X	
Data protection	X	X	X	X	X	X	X
Electronic mail	X	X	X	X			X
Expert systems					X	X	
Information systems	X	X	X	X		X	
Microforms	X	X					
Optical communication systems				X			X
Telecommunications	X	X	X	X	X	X	X
Cable TV			X	X		X	X
Computer controlled exchanges	X	X	X	X			
Satellite systems	X	X		X		X	X
Telephone networks	X	X	X	X	X	X	X
Video tape disk systems	X	X	X		X	X	X
Videotex and teletext	X	X	X	X			X
Voice communication with computers	X	X	X	X	X	X	X

Source: Information Technology–An Introduction, Peter Zorkoczy, 1982

Data networks

Computer communications have improved enormously since the first modems were introduced, transmitting data at rates around 2400 bits s^{-1}. Modems are now capable of data transmission rates up to 96 kbits s^{-1} and, in recent years, the concept of **networking** (i.e. the interconnection of large numbers of different types of data equipment) has shown many advantages over the PSTN. It is cheaper, faster and less prone to error.

Local area network (LAN) is the name given to a small network within one site, for example an office, factory or university. The **Cambridge Ring** is an example of such a network – it was originally developed for use within Cambridge University and has now been incorporated into an experimental office automation system linking word processors and data storage equipment. It can either



6. Local area network architectures: **(a)** Cambridge Ring; **(b)** Ethernet.

use a twin wire pair or coaxial cable and transmits data at a rate of 10 Mbits s^{-1} (*figure 6a*).

Ethernet has an open-ended structure as shown in *figure 6b*. The disadvantage of this kind of network is that there is no confirmation that a message has been received, and correcting transmission errors is difficult.

Other networks exist which can link equipment nationally or even internationally. Those come under the heading of **wide area networks**, or WANs, and include the European Euronet service and British Telecom's national data network, Packet SwitchStream (*see Communications 15*) which transmits data at 64 kbits s^{-1}. Additional information on WANs and LANs can be found in *Communications 18* and *20* respectively.

The impact of I.T. on society

The three strands of information technology – microelectronics, computers and telecommunications – are now so closely interrelated that they have a profound effect upon one another – technological advances in one area, immediately spill over into the other two. For example, improvements in either the processing power or data storage capabilities of solid state devices has an immediate impact on the computational speed and data handling capabilities of computers. Their range of applications therefore widens, for example, enabling faster switching centres for telecommunications exchanges to be developed.

It is this close association that has given rise to the new products and services of I.T., such as electronic mail, graphics systems, videotex, personal computers, video teleconferencing, optical communications systems, cable and satellite TV, data banks, electronic funds transfer, expert systems, and so on.

A second very important factor is falling costs. The constituent parts of such new products are becoming increasingly cheap to make, partly due to automated mass production techniques. New technology innovations are also being used in design and manufacture, reducing lead times to the introduction of a new component and hence also cutting costs. The result of this is a proliferation of products and services which manufacturers are striving to adapt to new and wider markets. Features such as touch screens and voice input and ouput (to be used alongside more conventional keyboards) serve to increase the attractiveness of new products to 'non-technical' people. Because of this, in the developed countries at any rate, we are witnessing the widespread infiltration of such products across the breadth of society.

For most of us, the most immediate affects have been felt in the home. Products such as home computers, video recorders and compact disc players have enjoyed enormous popularity since their introduction only two or three years ago. More conventional information related products, like the telephone and television and radio receivers, have also improved sales: there are now more than 423 million telephones in the world (28 million of them in the U.K.) and over 400 million television sets!

Concomitant with this has been the relative decline of the more traditional non-I.T. based information providers – the newspapers, cinema and books. Even at this level, we do appear to be moving towards an electronically based information society.

Initially, most I.T. products and services have been related to the office; the introduction of the word processor a few years ago has had the biggest single impact on office life. Now, innovations like personal executive computers employing electronic mail, digital PABXs (private automatic branch exchanges) which efficiently route all local and international calls around the building, teleconferencing and video teleconferencing are improving the

Below: computerised cash registers are now frequently used in shops. The assistant types in a code for each article and the price automatically appears on the display.

Science Photo Library/Jerry Mason

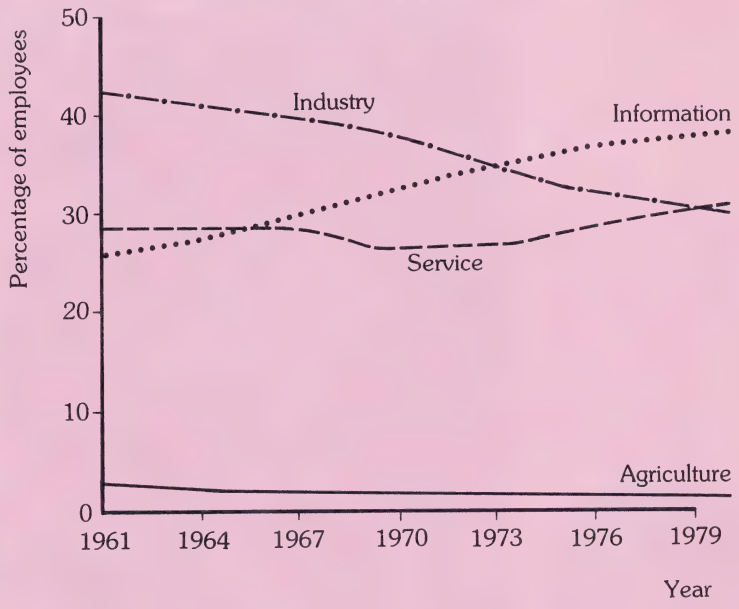
decision making machinery of management.

In both industry and commerce alike, distributed information systems based on local area networks are helping to quickly and efficiently channel this valuable commodity 'information' to those most in need of it. Indeed, one can plug into the office database via remote terminal and modem and also communicate verbally via cellular radio (see *Communications 11*) should you be away from the information source at any time.

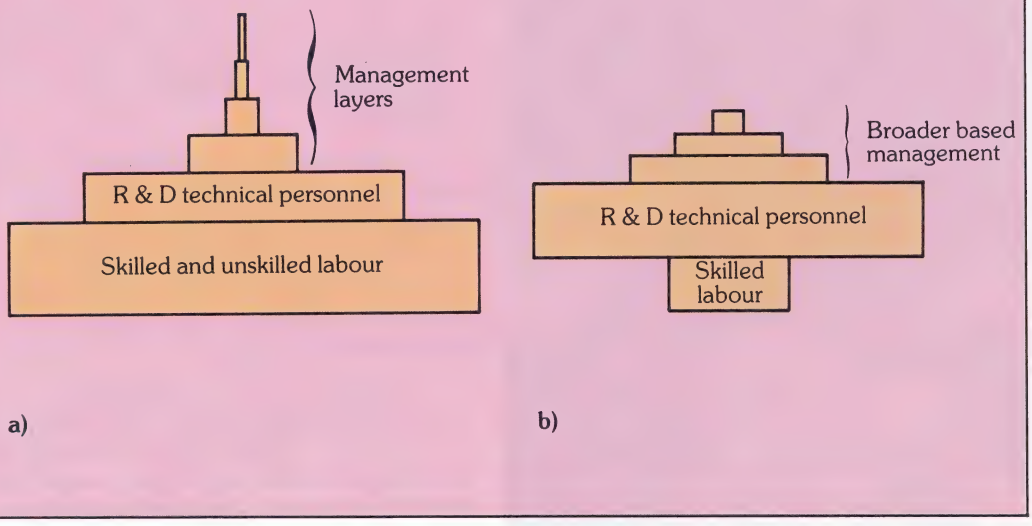
Employment

Undoubtedly, over the last 20 years there has been a marked increase in the percentage of the working population employed in information related and service industries, and a corresponding decrease in the numbers involved in the traditional 'wealth creating' manufacturing industries. This can be seen for the U.K. in the graph of *figure 7*. In some parts of the world, over 60% of the workforce is employed in the service sector.

7



8



7. The numbers of people employed in the information and services sectors of the economy has increased while employment in industry has fallen. *(Source: Information Technology – An Introduction, Peter Zorkoczy, 1982).*

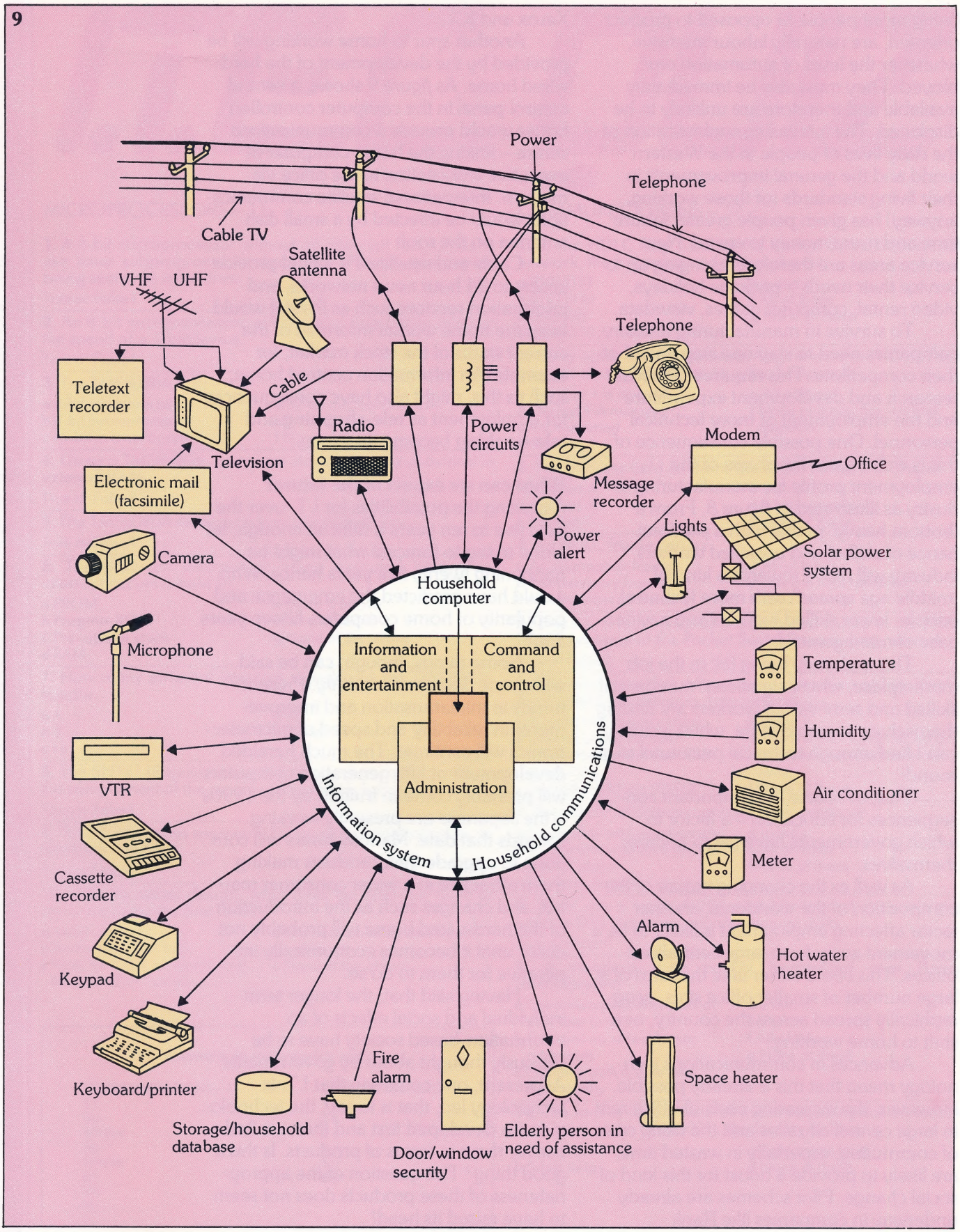
8. The changing shape of the employment profile.

9. The 'home of the future?' The computer acts as a central controller and communications centre. *(Source: Information Technology – An Introduction, Peter Zorkoczy, 1982).*

Of course, it must not be assumed that these trends are solely the result of advances in I.T. Certainly, automation in some industries has necessitated the shedding of both skilled and unskilled labour, but equally important has been the shift of manufacturing industry to those countries, like Korea, Singapore, Malaysia and Hong Kong, where labour costs are lower.

Service industries on the other hand,

9


Power
Telephone
Cable TV
Satellite antenna
VHF UHF
Telephone
Teletext recorder
Cable
Radio
Power circuits
Modem
Office
Television
Electronic mail (facsimile)
Message recorder
Power alert
Lights
Solar power system
Camera
Household computer
Information and entertainment
Command and control
Microphone
Temperature
Humidity
VTR
Administration
Air conditioner
Information system
Household communications
Meter
Cassette recorder
Alarm
Hot water heater
Keypad
Keyboard/printer
Fire alarm
Space heater
Storage/household database
Door/window security
Elderly person in need of assistance

being more people as opposed to product oriented, are naturally labour intensive, whatever the level of automation employed. They must also be immediately available and therefore are unlikely to be displaced. The increasing sophistication of the daily lives of people in the Western world and the general improvements in their living standards (of those working, anyway) has given people greater leisure time and more money to spend. New service areas are therefore springing up to service their needs – package holidays, video rental, computer games, viewdata.

To survive in manufacturing industry, companies need to stay one step ahead of their competitors. This requires significant research and development expenditure and the employment of more technical personnel. One possible consequence of this is a change in the shape of the employment profile for manufacturing industry as illustrated in *figure 8*. From a 'bottom heavy' profile with a predominance of skilled and unskilled workers, industry will move towards a kind of 'middle age spread' with more technical people, fewer skilled workers and a wider base of management.

This is already being felt in the job marketplace, where significant numbers of skilled and semi-skilled workers are finding themselves unemployable, whilst industry can't find enough technical personnel to go round!

This, of course, has important consequences for education – a factor to which governments have yet to address themselves.

As well as the changing nature of the composition of the workforce, another factor affecting employment is likely to be a movement away from large centralised offices. This could either take the form of a large number of smaller office sites, geographically spread across the country, or a shift to home working.

Advances in communications technology mean that this is already possible. However, the increasing costs of office rent in large central city sites and the rising costs of commuting, especially in wasted time, are likely to provide a boost for this kind of social change. Pilot schemes are already underway in companies like Rank Xerox and ICL.

Another spur to home working will be provided by the development of the hard-wired home. As *figure 9* shows, a central control panel in the computer controlled house would provide a communications centre – linking the home computer or executive workstation to the office via modem. International satellite communications would be effected by a small dish antenna on the roof.

Cable and satellite TV would provide access to 24 hour news networks, and information services such as Prestel would keep the home worker informed of the current status of the stock market, for example. An information centred home, such as this, might also have consequences for employment as **tele-shopping** and **tele-banking** become the norm.

What can we expect of the future?

Assessing the possibilities for I.T. over the next five to ten years is difficult enough, let alone trying to forecast what might be happening twenty five years hence. Who would have predicted the emergence and popularity of home computers fifteen years ago?

Some things, though, can be said with some degree of certainty. Present trends in miniaturisation and improvements in reliability and speed of microelectronics will continue. The much heralded development of fifth generation computers will probably come to fruition by the 1990s – the Japanese are presently working towards that date. Manufacturers will continue to upgrade their products making them attractive to a wider consumer market, and changes such as the introduction of the hard-wired home will probably not occur until it becomes economically imperative for them to do so.

Having said that, the longer term individual and social effects of an information-based society have to be seriously thought about by governments. At present, one could say that I.T. is technology led, that is to say, the technologies are developed first and then we find a use for them in terms of products. Is this a good thing? The question of the appropriateness of these products does not seem to have raised its head!

TEST YOUR PROGRESS with the

I.T.E.C. QUIZ

MICROPROCESSORS – 11

1. A 4-bit microprocessor without an interrupt facility has to use timer subprograms to check for external information being sent to its input lines.

True or False?

2. An 8-bit microprocessor will perform operations at twice the speed of a 4-bit device.

True or False?

3. The TMS8080A is a:

a 4-bit microprocessor
b 8-bit microprocessor
c 16-bit microprocessor
d 32-bit microprocessor
e Transputer
f None of these

4. Using a TMS8080A, a register stack has to be provided in external RAM.

True or False?

5. The TMS8080A receives interrupts on its EI pin.

True or False?

6. The program information being executed before an interrupt occurs is held in:

a PROM
b Program ROM
c The register stack
d RAM
e a and b above
f c and d above
g None of the above

7. How many different interrupt signals can the TMS8080A handle:

a 4
b 8
c 16
d 80
e None of these

8. The SN74148 generates a 3-bit code that corresponds to the complement of the input number that is brought low.

True or False?

9. The TIL308 seven segment LED display is controlled by its blanking input (BI).

True or False?

10. With the sorter design, what does the operator have to do to start it for the first time?

a Switch it on
b Set the RESET FF to 0
c Press the DISPLAY switch and then the RESET switch
d Connect a battery and stand well back
e None of the above

Answers to last week's quiz

COMMUNICATIONS – 16	BASIC REFRESHERS
1 c	**1** e
2 a	**2** e
3 False	**3** d
4 b	**4** d
5 False	**5** a
6 f	**6** a
7 b	
8 c	

COMMUNICATIONS – 17

1. In the HDLC flag field of the Packet SwitchStream data network, the following code(s) must exist:

a 01010101
b 01111110
c 10000001
d 10111110
e b and d
f All of these
g None of these

2. If any batch of data between flag fields of an HDLC frame consists of a string of five 1s, what happens?

a A further 1 is inserted
b The last 1 is removed
c A 0 is inserted
d a and c
e All of these
f None of these, this is a trick question

3. The address field contains the code: 00000111. This is:

a Address A
b Address B
c Address C
d a or b
e A trick question
f None of these

4. An HDLC frame can be:

a An information frame
b A supervisory frame
c An unnumbered frame
d All of these
e This is a trick question
f None of these

5. There are four types of unnumbered frame in the HDLC level 2 protocol of the Packet SwitchStream data network.

True or False?

6. The process which allows error detection at link level in the Packet SwitchStream data network is:

a A cyclic redundancy check
b Useless unless the user information is less than 128 octets long
c Given the address 00000011
d a and b
e This is a trick question, error detection occurs at network level
f None of these

7. Packet SwitchStream network level protocol:

a Has end-to-end significance
b Is sometimes called a packet level protocol
c Concerns level 3
d a and c
e All of these
f None of these

8. There are a total of 256 possible logical channels over which a DTE may transmit and receive data.

True or False?

9. A standard data packet transmitted over Packet SwitchStream:

a Contains a packet header of only four octets
b Contains no network user addresses
c Contains both a send and receive tally
d a and b
e b and c
f All of these
g None of these

10. The facility length field of a call request packet contains the binary code: 00000001. The facility field is:

a Not transmitted
b 8-bits long
c Eight octets long
d None of these

COMING IN PART 48

What advantage does a **16-bit microprocessor** have over a 4 or 8-bit device? *Microprocessors 12* finds out by looking at the TMS9980A.

Communications 18 takes a look at **Wide Area Networks** or WANs.

The **TM 990/U89 microcomputer module** is a self-contained computer. *Microprocessors 13* discusses one possible application for it.

What is an **SPC PABX**? And how do you use a PABX in a **data network**? Find out in *Communications 19*.

PLUS: *Basic Theory Refresher* – which takes a look at the consequences of **incorrectly terminated transmission lines**.